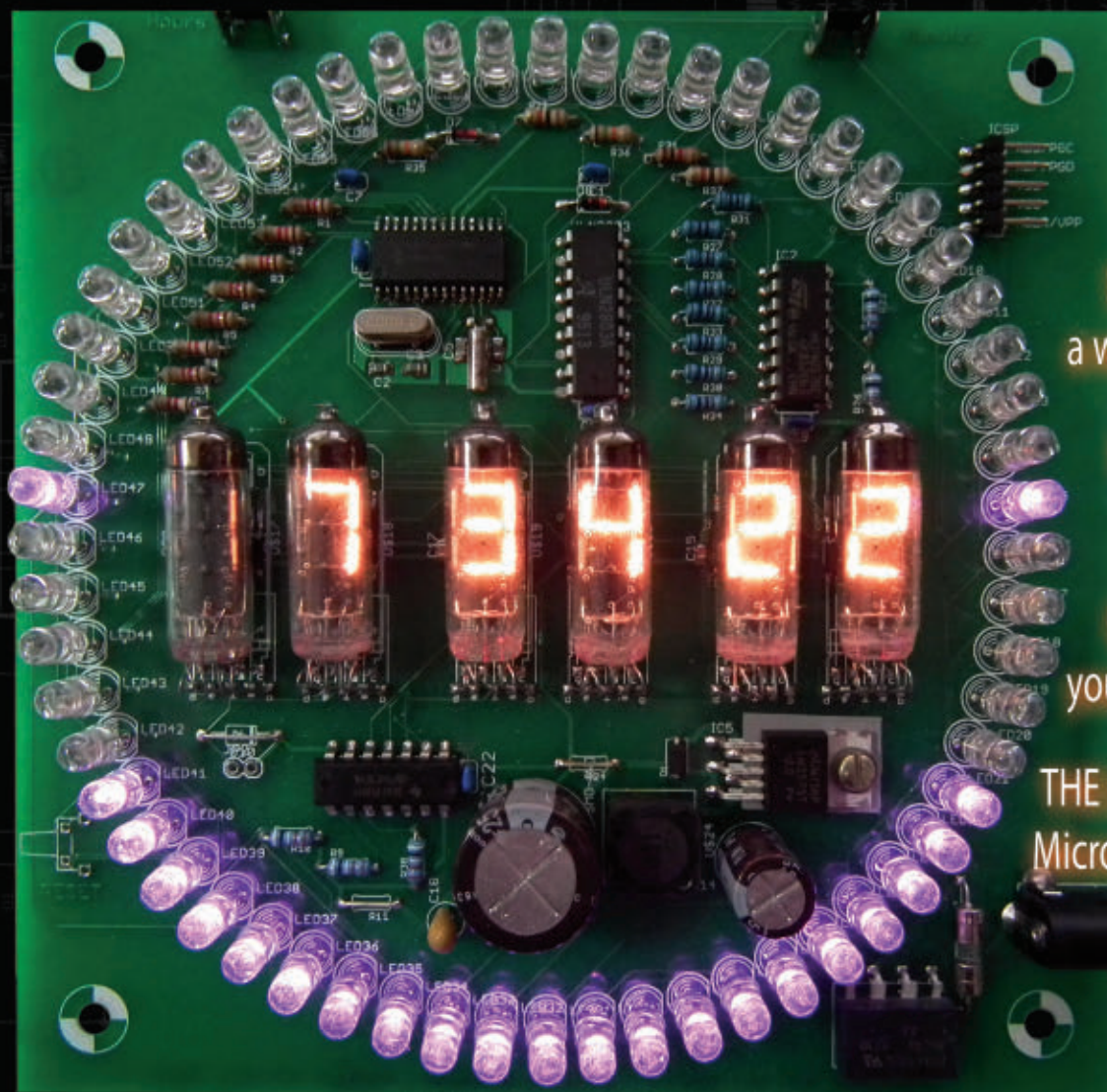


PROJECTS THEORY APPLICATIONS CIRCUITS TECHNOLOGY

NUTS AND VOLTS

www.nutsvolts.com
September 2016

EVERYTHING FOR ELECTRONICS



ESCAPE ROOM
DIY electronic lock

SUBSTITUTION BOXES
a workbench must have!


HACKING THE AMIGO
Retrokit Computer

COMMON ANTENNAS
you need to know about

THE MICROCHIP PIC32MM
Microcontroller - First Look

Build The
NUMITRON CLOCK

TS-7680
WITH **TS-SILO**
TECHNOLOGY FOR UP TO

6  **SECONDS**
POWER HOLD

-40°C TO +85°C
OPERATING TEMPS



32 YEARS
MADE IN USA

35G
VIBRATION
IN RECENT
HALT

2X USB
CAN
RS-485
10/100

454 MHZ
NXP
ARM
CPU

802.11 BGN 
WIFI
BLUETOOTH 4.0+ EDR

\$159
QTY 100

2 SEC
BOOT TIME

 **8 TO 40 VDC**
10 TO 28 VAC

Need a Scope?

Low Prices - Free Support - Excellent Customer Service



Alan Lowne, Saelig CEO, says: "Many questions arise when choosing a new digital oscilloscope. And it's a decision you don't want to regret! A first place to turn is often the plethora of reviews on YouTube – they can show you more details than just the specs. Google the product and search for videos. Then think about where will you use the scope (on the bench, at a customer's site, under the hood of a car.) How many signals do you need to measure at once? What are the maximum and minimum amplitudes of signals that you need to measure? What is the highest frequency of signal you need to measure? Are your signals repetitive or single shot? Do you need to view signals in the frequency domain (spectrum analysis) as well as the time domain? Cost is always a factor too. Memory depth – do you need to zoom in on small signal details? So look at these criteria..."

Form Factor – traditional bench-top, hand-held, or PC-based?

Bandwidth – for square-waves you'll need a scope with 5x higher bandwidth than the signal frequency.

Sample Rate – often depends on how many channels are in use.

Waveform Capture Rate – faster the better (defines the "dead space" of missed signals).

Memory Depth – a large memory will let you zoom in on small, fast, infrequent glitches.

Resolution and Accuracy – most scopes are 8-bit; 12-bit is great for seeing tiny signal changes

Triggering Capabilities – check if you need something special like triggering on digital waveform patterns

Input Ranges (& Probes) – typical scopes are +/-50mV to +/-50V

Connectivity – need remote access? USB data storage? WiFi access?

Built-in Capabilities – automatic measurements, pass/fail etc.

Ease of Use – "one-touch" automatic setup, memorized configurations, awkward multiple menu steps?

MSO Ready? – will you need simultaneous digital bus debugging as well as analog signal capture?

AWG? – a built-in signal generator saves space and is portable but may have limitations.

100 MHz Economy Oscilloscope Comparison Chart

	Siglent SDS1102CML	Owon SDS7102V	Rigol DS1102E	Siglent SDS1102X	Rigol DS1104ZPlus	Teledyne LeCroy WaveAce 1012	Siglent SHS810	PicoScope 3205D
Format	Benchtop	Benchtop	Benchtop	Benchtop	Benchtop	Benchtop	Handheld	PC-Based
Channels	2	2	2	2	4	2	2	2
Bandwidth	100 MHz	100 MHz	100 MHz	100 MHz	100 MHz	100 MHz	100 MHz	100 MHz
Max Sample Rate	1 GSa/s	1 GSa/s	1 GSa/s	1 GSa/s	1 GSa/s	500 MSa/s	1 GSa/s	1 GSa/s
Waveform Capture Rate	60,000 wfm/s	40wfm/s	?	60,000 wfm/s	30,000 wfm/s	?	?	80,000 wfm/s
Memory	2Mpts	10Mpts	1Mpts	14Mpts	12Mpts	2Mpts	2Mpts	256Mpts
AWG?	Optional	No	No	Optional	Optional	No	No	Yes
MSO Ready?	No	No	No	Yes	Yes	No	No	No
Display Size	8" WVGA	8" WVGA	5.7" QVGA	8" WVGA	7" WVGA	7" WVGA	5.7" QVGA	N/A
Standard Triggers	Edge, Pulse Width, Slope, Video, Alt	Edge, Pulse Width, Slope, Video, Alt	Edge, pulse width, slope, video, pattern, continuous time, Alt	Edge, Pulse Width, Slope, Video, Pattern, Duration	Edge, Pulse Width, Slope, Video, Pattern, Duration	Edge, Pulse Width, Slope, Video, Alt	Edge, Pulse Width, Slope, Video, Alt	Edge, window, pulse width, window pulse width, dropout, window dropout, interval, logic, runt pulse
Serial Decode	Optional	No	No	Optional	Optional	No	Optional	Yes
Measurements	32	19	20	33	33	32	33	30
Battery?	No	Optional	No	No	No	No	Yes	No
Math Functions	A+B, A-B, AxB, A/B, FFT	A+B, A-B, AxB, A/B, FFT	A+B, A-B, AxB, FFT	A+B, A-B, AxB, A/B, FFT, integration, differentiation, square root	A+B, A-B, AxB, A/B, FFT, AND, OR, NOT, XOR, Intg, Diff, Sqrt, Lg, Ln, Exp, Abs	A+B, A-B, AxB, A/B, FFT	A+B, A-B, AxB, A/B, FFT	-A, A+B, A-B, A'B, A/B, A^B, sqrt, exp, ln, log, abs, Norm, sign, sin, cos, tan, arcsin, arccos, arctan, sinh, cosh, tanh, derivative, integral, delay
Price	\$319.00	\$389.00	\$399.00	\$499.00	\$879.00	\$960.00	\$573.00	\$892.48

Economical scopes are now available with capabilities that rival the big name manufacturers at well below \$1,000. Note that bandwidth and sampling rate are not upgrade options on most DSOs, so once you've bought your product of choice you are stuck with your decision. "Hacking" upgrades are not recommended as they void a manufacturer's warranty! At Saelig Co. Inc. we have assembled the widest range of affordable scope solutions, from low-end USB scope adapters at under \$120, to sophisticated yet economical standalone scopes, to high-end 12-bit 2/4-channel mixed-signal scopes that cover 1GHz signals, as well as offering 8/16 channels of simultaneous logic analysis, and even up to the world's-fastest 12GHz sampling scope adapter. Details at <http://www.saelig.com/category/PS.htm>



• Saelig Company, Inc. • www.saelig.com • sales@saelig.com • 1-888-7SAELIG •

September 2016

26 Build the Numitron: A Six-Digit Clock

Don't miss out on this opportunity to build a useful project that showcases cold war era components in a beautifully designed timepiece.

■ By Bill van Dijk

35 Vintage Computing — Simple Hardware Interfaces for Your Mentor's Friend

Previous articles here introduced the Mentor's Friend: a Propeller-based retro computer that you and a young protégé can build together and program in BASIC. Most of the projects in these initial articles focused on the fundamentals (with emphasis on "fun") of Color BASIC which is the software "operating system" inside the Amigo retro computer. This time, we'll venture outside Color BASIC to interface with a couple of simple hardware circuits.

■ By Dane Weston

62 Name That Part!

Think you know your stuff? Try this month's quiz that will test your memory and your wits.

■ By David Goodsell

08 TechKnowledgey 2016 Events, Advances, and News

Read about a planar lens for smartphones; an optical clock accurate in the attosecond range; the new Asus' ZenBook 3 laptop; how to run a quantum computer; what to do if you really need a lot of storage; a supply for your Pi; never losing anything again; and what happened to Bell Labs.

12 Q&A

Reader Questions Answered Here

Get some enlightenment on modern lighting and whether green energy is truly "green."

18 The Ham's Wireless Workbench

Practical Technology from the Ham World

Common Antennas You Need to Know About

In today's wireless communication equipment, there has to be an antenna. It might be hidden, it might be tiny, but it's there. Knowing about common antennas is a good step toward effective data link and other wireless system design.

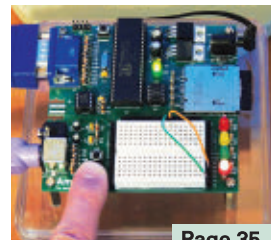
41 Hail, the Lowly Substitution Box!

In decades past, capacitor and resistor substitution boxes were very popular pieces of test equipment. These days, it seems folks have forgotten their value and ease of use. Here's a discussion on how they work and the different options and styles available, so you can start using them for yourself.

■ By Robert Reed



Page 41



Page 35

Departments

05 DEVELOPING PERSPECTIVES

So, Exactly What is Electronics?

06 READER FEEDBACK

24 NEW PRODUCTS

25 SHOWCASE

NV WEBSTORE

56 CLASSIFIEDS

64 TECH FORUM

66 ELECTRO-NET

66 AD INDEX

50 The Design Cycle

Advanced Techniques for Design Engineers

Help is Finally Here for 32-bit PICs

Have you been sitting on the eight-bit/32-bit microcontroller fence wondering if you're ever going to be able to fall on the 32-bit side? Well, now is a good time to tilt towards 32-bit territory. Microchip has finally released a 32-bit microcontroller that is supported by the MPLAB Code Configurator. If the new low cost/low pin count 32-bit PICs don't push you over to the 32-bit side, the free tools and 32-bit code generation capabilities of MPLAB Code Configurator will.

57 The Spin Zone

Adventures in Propeller Programming

Game Time

We've been talking recently about escape rooms and the elements needed to create them. This time, we'll knit these parts all together to create a timed game piece that can be reconfigured from its own interface, and that can be implemented in other applications as well.

Published Monthly By
T & L Publications, Inc.

430 Princeland Ct.
Corona, CA 92879-1300
(951) 371-8497

FAX **(951) 371-3052**

Webstore orders only **1-800-783-4624**

www.nutsvolts.com

Subscription Orders

Toll Free **1-877-525-2539**

Outside US **1-818-487-4545**

P.O. Box 15277

North Hollywood, CA 91615

FOUNDER

Jack Lemieux

PUBLISHER

Larry Lemieux

publisher@nutsvolts.com

ASSOCIATE PUBLISHER/ ADVERTISING SALES

Robin Lemieux

robin@nutsvolts.com

EDITOR

Bryan Bergeron

techedit-nutsvolts@yahoo.com

VP OF OPERATIONS

Vern Graner

vern@nutsvolts.com

CONTRIBUTING EDITORS

Tim Brown

Fred Eady

Jon McPhalen

Jeff Eckert

Ward Silver

David Goodsell

Robert Reed

Dane Weston

Bill van Dijk

CIRCULATION DEPARTMENT

subscribe@nutsvolts.com

SHOW COORDINATOR

Audrey Lemieux

WEBSTORE MARKETING COVER GRAPHICS

Brian Kirkpatrick

sales@nutsvolts.com

WEBSTORE MANAGER/ PRODUCTION

Sean Lemieux

sean@nutsvolts.com

ADMINISTRATIVE STAFF

Re Gandara

Copyright © 2016 by T & L Publications, Inc.

All Rights Reserved

All advertising is subject to publisher's approval. We are not responsible for mistakes, misprints, or typographical errors. *Nuts & Volts Magazine* assumes no responsibility for the availability or condition of advertised items or for the honesty of the advertiser. The publisher makes no claims for the legality of any item advertised in *Nuts & Volts*. This is the sole responsibility of the advertiser. Advertisers and their agencies agree to indemnify and protect the publisher from any and all claims, action, or expense arising from advertising placed in *Nuts & Volts*. Please send all editorial correspondence, UPS, overnight mail, and artwork to: **430 Princeland Court, Corona, CA 92879**.

Subscription Information

Nuts & Volts — PO Box 15277, North Hollywood, CA 91615-9218

Call **877-525-2539** or go to **www.nutsvolts.com**

Subscribe • Gift • Renewal • Change of Info

DEVELOPING PERSPECTIVES

by
**Bryan
Bergeron,**
Editor

So, Exactly What is Electronics?

Presumably, if you're reading this, you're at least curious about electronics. If you're a first time visitor to *Nuts & Volts*, perhaps something on the cover caught your eye. If you're a long time subscriber, the content hopefully resonates with what you enjoy reading to advance your understanding and practice of electronics. However, given the rapid evolution of the field over the past few years, exactly how do you define electronics today?

As a point of reference, according to **Merim-Webster.com**, electronics is "a branch of physics that deals with the emission, behavior, and effects of electrons (as in electron tubes and transistors) and with electronic devices." I don't know about you, but to me, this definition seems both indefinite and antiquated. In this broad definition, "electronics devices" includes just about everything — from resistors to Bluetooth enabled bathroom scales to ham radio transceivers. This definition also ignores the past several decades of fundamental electronic engineering teachings that include the behavior and effect of holes as well as electrons.

This dictionary definition of electronics highlights the nature of the electronics hobby today. For one, it's extremely diverse. As active hobbyists, we can be talking about consumer electronics, robotics, medical devices, or video game consoles. The definition also hints of experimentation or at least appreciation of the underlying physics.

As such, I believe an electronics hobbyist isn't a label for someone who simply purchases the latest smartphone or game console, but someone who also digs into how the devices work. I see lots of people with their phones, PDAs, laptops,

fitness bands, and the like, and I don't consider them electronics hobbyists. They're consumers.

In my definition of electronics, there's definitely a hands-on component, and it's central to the definition. Someone may own all of the latest electronics gizmos, but if they don't use a soldering iron or equivalent connection system at least once a month, then I'd be hard-pressed to acknowledge that they're an electronics enthusiast. In my definition, someone active in electronics pursues active investigation.

I say active because you can't simply buy yourself into competence. You may have the best stocked parts bin on the planet, but if you don't use the parts in circuits, you might as well sell the parts collection. As a ham radio operator, I've seen many "shack" photos some showing dozens of transceivers, tuners, 24-hour clocks, and so on. Very impressive, but I know that the ham radio operator probably uses only one of the transceivers on a regular basis, and the rest is just window dressing. As a ham radio operator with a few extra communications peripherals of my own, I can vouch for the attraction of vintage gear that simply looks beautiful.

Based on the evolution of components to microcontrollers over the past few years, it's not unreasonable that *Nuts & Volts* may have to change its name to *Keyboards & Microcontrollers*. Enough of my rant. I'd like to hear from you.

What's your definition of electronics? Does it have to include a soldering iron? Does programming DSP chips and other keyboard activity qualify? Where do you see electronics going in, say, 10 years?

NV

READER FEEDBACK

Feelin' Pumped

I loved Joe Bidwell's article in the July issue of *Nuts & Volts* regarding the water pump protection system! It opened my eyes to control devices that I never knew existed. I have a few questions I am hoping Joe would help me with.

There is a box called the "pump control" that is mentioned several times. Why is it necessary and what is its function?

In the article, Joe says the pump current can be checked at the pump control box while the pump is running. How do you do that?

I would like to get Joe's opinion. Instead of purchasing separate units for the voltage relay, time-out relay, and time-delay relay, what would you think of building a single device from scratch using a PIC microcontroller driving a single relay? Other than design/construction time savings, are there any advantages to spending the dollars to purchase the commercially manufactured units?

Judy May W1ORO
Union, KY

I'm glad that you enjoyed the article, Judy. I'll do my best to answer your questions. Pump Control Box: Single-phase pumps in this range come in two varieties: "two-wire" and "three-wire." (There is also a ground wire, so

actually either three or four wires are fed down to the pump). All of these pumps have separate Start and Run windings. The Start winding gets a momentary voltage through the start relay and capacitor. The two-wire pumps have this built into the motor, so a control box is not needed. They can run right from the pressure switch. My pump is a three-wire that brings the Start winding up separately from the Run winding. The Control Box just houses the start relay and capacitor. Hope that explains it.


Measuring Current: If you look at the bottom left of Figure 9, you'll see a yellow wire that loops back into the control box. This is one of the 240V lines that runs the pump. I looped it so I could use a clamp-on ammeter to check current. You can really do this on either of the 240V lines anywhere on your system.

PIC Microcontroller: I don't know why you couldn't do all of these functions with a PIC. I personally don't have much experience with them. I took the hardware approach mainly because I started out with the voltage relay which I found on a 'surplus' shelf for \$10. My concern with a PIC approach would be with the voltage spikes generated by motor starts, and even the contactor relay energizing. Maybe that is not an issue, just my opinion.

Thanks for your note.

Joe Bidwell KF7ODK

The Easiest Way to Design Custom
Front Panels & Enclosures



Free Front Panel Designer

You design it
to your specifications using our FREE CAD software, Front Panel Designer

We machine it
and ship to you a professionally finished product, no minimum quantity required

- Cost effective prototypes and production runs with no setup charges
- Powder-coated and anodized finishes in various colors
- Select from aluminum, acrylic or provide your own material
- Standard lead time in 5 days or express manufacturing in 3 or 1 days

FRONT PANEL EXPRESS
FrontPanelExpress.com



AP CIRCUITS
PCB Fabrication Since 1984

As low as...

\$9.95
each!

Two Boards
Two Layers
Two Masks
One Legend

Unmasked boards ship next day!

www.apcircuits.com



Sounding Off

Regarding the Developing Perspectives editorial about The Art of Electronics, my interest has been advocating better audio quality of electronics. No, I don't mean advancing the state of the art with more costly and exotic equipment, but improving what all of us hear regularly; 99.5% of the electronics I hear sound very unnatural, having a narrow frequency response and noticable distortion. Additionally, most products have an obnoxious resonant peak between 150 and 300 Hz; perhaps purposely to give the untrained ear the impression of more bass response than really exists. To me, it sounds very hollow and drummy. Could someone please develop a standard of audio reproduction decency which would provide a clear and natural rendition of human voices and most musical sounds (perhaps a smooth response from 80 Hz-12 kHz, with distortion low enough to be unobtrusive to a critical listener hearing it blindly), then develop and produce something which would accomplish this as simply and economically as possible? Even monophonic versions could be offered if cost or size are limited, with one better channel of sound instead of multiple poor quality channels.

Probably the biggest obstacle to this is the needed speaker and enclosure size. At 80 Hz, a wavelength of sound through air is almost 14 feet — so much air must be

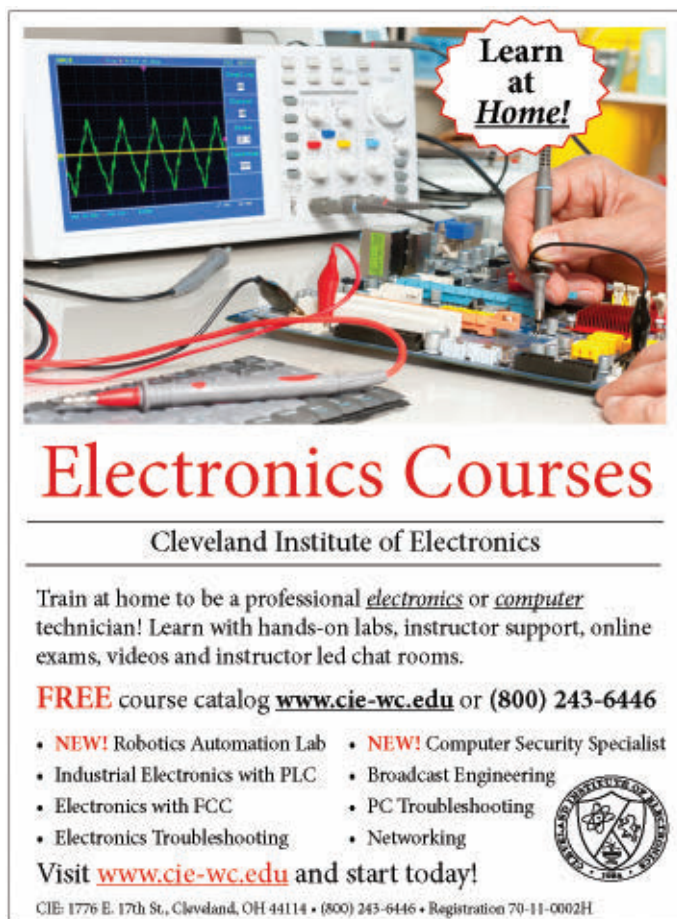
moved to produce it. As a result, the speaker would need to be several inches in diameter, and the cabinet which houses it may need to be about a cubic foot in volume. This would also need to be quite rigidly constructed to prevent resonant vibrations which would unnaturally color the sound. Understandably, this would not be possible in portable devices (Is this why earphones are so popular now?), but should be reasonable with stationary systems. The general quality of sound can be improved; it is mainly a matter of knowing and caring. While "Wi-Fi" is becoming all the rage, "Hi-Fi" — from which Wi-Fi was imitated — is dying.

Michael Kiley

First, thanks for taking the time to write. I hear you, and blame it on the popularity of the iPad/iPhone as playback devices. At some point, it became more important — from a marketing perspective — to have a device with, say, 1,000 songs than with 100 songs of superb quality. Decades ago, I used to set up listening rooms, complete with plush carpet on the floor, fiberglass-filled frames on the walls, and physically large speakers. The best setup I ever did used refrigerator-sized theater speaker assemblies. Ah, the good old days ...

Perhaps your note will stir others to action.

Bryan Bergeron



Learn at Home!

Electronics Courses

Cleveland Institute of Electronics

Train at home to be a professional electronics or computer technician! Learn with hands-on labs, instructor support, online exams, videos and instructor led chat rooms.

FREE course catalog www.cie-wc.edu or (800) 243-6446

- **NEW!** Robotics Automation Lab
- **NEW!** Computer Security Specialist
- Industrial Electronics with PLC
- Broadcast Engineering
- Electronics with FCC
- PC Troubleshooting
- Electronics Troubleshooting
- Networking

Visit www.cie-wc.edu and start today!

CIE: 1776 E. 17th St., Cleveland, OH 44114 • (800) 243-6446 • Registration 70-11-0002H



WORLD'S MOST VERSATILE CIRCUIT BOARD HOLDERS

Model 324

Our Circuit Board Holders add versatility & precision to your DIY electronics project. Solder, assemble & organize with ease.

VISIT US ON   

MONTHLY CONTEST

Visit us on Facebook® to post a photo of your creative PanaVise project for a chance to win a PanaVise prize package.

Model 201

PANA VISE®

Innovative Holding Solutions

7540 Colbert Drive • Reno • Nevada 89511 | (800) 759-7535 | www.PanaVise.com

ADVANCED TECHNOLOGY

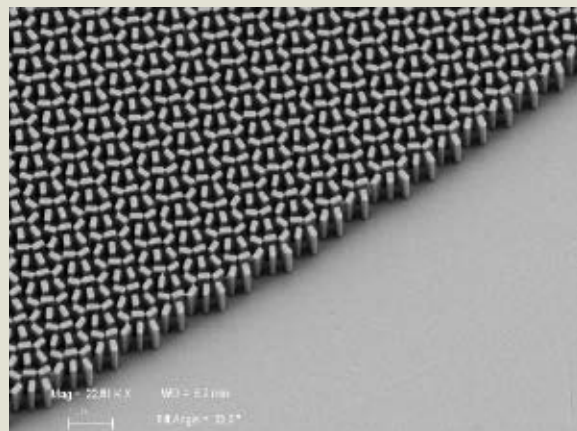
Planar Lens for Smartphones

The lens on your smartphone's camera isn't exactly huge, but like the lenses in SLRs, telescopes, and microscopes, it uses a stack of curved optics to provide focus, reduce distortion, and provide a good clear image. However, some researchers at Harvard's Paulson School of Engineering and Applied Sciences (www.seas.harvard.edu) found themselves asking, "What if you could replace those stacks with a single flat — or planar — lens?" As it turns out, you can.

As proof, they recently demonstrated "the first planar lens that works with high efficiency within the visible spectrum of light, covering the whole range of colors from red to blue." It can resolve features down to nanoscale size that are separated by less than the wavelength of light.

Instead of using curved glass or plastic to bend light, it employs an ultra-thin array of tiny waveguides, referred to as a metasurface. According to Prof. Federico Capasso, "This technology is potentially revolutionary because it works in the visible spectrum, which means it has the capacity to replace lenses in all kinds of devices; from microscopes to cameras, to displays and cell phones. In the near future, metalenses will be manufactured on a large scale at a small fraction of the cost of conventional lenses, using the foundries that mass produce microprocessors and memory chips."

To focus light in the visible range, the Paulson team needed to find a material that wouldn't absorb or scatter light, and that would strongly confine light with a high refractive



■ Scanning electron microscope image of a metalens.

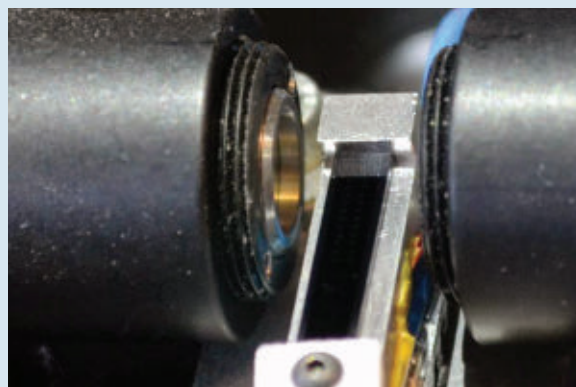
index. The answer turned out to be titanium dioxide: a common material that is used in things like paint and sunscreen. From this material, they created an array of nanostructures to form the heart of the metalens.

The result is a planar lens that can resolve items as small as 400 nm across. This approach should also address weight, size, power, and cost issues that need to be solved for future high performance augmented reality/virtual reality (AR/VR) wearable displays. The researchers have filed for patent protection and are presently looking for commercial development opportunities. ▲

Optical Clock Accurate in Attosecond Range

The most accurate timepieces in use today are atomic clocks, commonly employed to keep time for Internet and satellite communications and various astronomical purposes. Their operation is based on naturally occurring frequencies of atoms responding to radiation. This works pretty well, providing accuracy to about 1×10^{-13} second, or a tenth of a trillionth of a second. Apparently, some people are just never satisfied, however, including researchers at UCLA's Henry Samueli School of Engineering and Applied Science (engineering.ucla.edu). Last May, they demonstrated an optical clock that can track time with precision to 270 quintillionths of a second; a quintillionth being 1×10^{-18} , or 0.000000000000000001. This is known as an attosecond. For reference, one attosecond is to one second as one second is to about 31.71 billion years.

Optical clocks have been around for several years, but previous models were much larger than the new one, as they used large fiber lasers that needed to be housed in enclosures about the size of a desktop computer. The UCLA team was able to shrink the mechanism to about 1 cc using



■ UCLA's optical clock (the small, black strip between the cylinders).

a process similar to silicon chip fabrication.

According to the team, the new clock may lead to more precise measurements of space and time, and could have applications in optical, wireless, and space based communications. For example, it could be used to measure the movement of atoms or to discern the movement of distant objects far beyond our solar system. ▲

COMPUTERS and NETWORKING

Meditate on This

Laptop designers seem to believe that buyers prize thinness above all other features, and Asus' new ZenBook 3 delivers the goods in that category. The machine is shoehorned into a case that is only 11.9 mm (0.47 in) thick, which is accomplished by using an aerospace-grade aluminum alloy that is said to be 50 percent stronger than standard materials. This is a little fatter than the HP Spectre, which slides in at 10.4 mm (0.409 in), but the ZenBook wins in the weight category at 910 g (32.1 oz) vs. 1100 g (38.8 oz) — possibly because of the smaller display (12.5 in vs. 13 in).

Behind the Gorilla Glass 4 screen, you can equip it with an Intel Core i7 processor, 16 GB of 2,133 MHz RAM, and up to as much as 1 TB of SSD storage. It also comes with a USB Type-C port and a four-speaker Harman Kardon sound system. Included is a full size backlit keyboard and a glass covered touchpad that incorporates palm-rejection technology, Smart Gestures, and handwriting support, plus it has a fingerprint reader to eliminate password access.

According to Asus, the ZenBook provides up to nine hours of battery life and recharges to 60 percent capacity in only 49 min. The maxed-out machine will run you \$1,999, but if you scale back to an i5 processor, 4 GB of RAM, and a 256 GB drive, you can squeak in at about half that price. ▲



■ The Asus ZenBook 3 is less than 0.5 in thick and weighs only 32.1 oz.

Run a Quantum Computer

If you have ever wanted to learn more about quantum computing, now's the time. All you have to do is access the IBM Quantum Experience at www.research.ibm.com/quantum, and you will have access to a cloud-enabled quantum computing platform where you can "run algorithms and experiments on IBM's quantum processor, work with the individual quantum bits (qubits), and explore tutorials and simulations around what might be possible with quantum computing."

The quantum processor — housed at the T.J. Watson Research Center in New York — is made up of five superconducting quantum bits. This doesn't sound like much, but IBM plans on scaling it up to larger systems, leading to the establishment of a universal quantum computer with a processor having 50 to 100 qubits sometime in the next decade. Eventually, IBM envisions a machine with more than 100,000 physical qubits. Even with only 50 qubits, such a device would be superior to all of today's TOP500 supercomputers.

It's a pretty murky concept, but the fundamental principle is that, unlike a classic bit that represents either a one or a zero, a qubit can represent a one, a zero, or both at once. From there, it gets more complicated, but the website offers a primer on quantum computing that will help you get started.

IBM notes, "By giving users access to the IBM Quantum Experience, it will help businesses and organizations begin to understand the technology's potential, for universities to grow their teaching programs in quantum computing and related subjects, and for students to become aware of promising new career paths." ▲



■ An IBM quantum computing scientist uses a tablet to interact with a quantum computing platform.

COMPUTERS and NETWORKING Continued

If You Really, Really Need a Lot of Storage

Most people will never need this much drive space, but the new storage unit is worth a look, even if only as an item of curiosity. It was designed by Neil Poulton, an award-winning product designer known for “deceptively simple looking mass-produced objects.” The unit is a 12-bay desktop monster that can store up to 96 TB of data, which is said to be 50 percent more than any other desktop directly attached product on the market.

LaCie — which is Seagate’s premium brand — has equipped it with a hardware RAID controller and 7,200 RPM drives, which provides speeds of up to 2,600 MB/s. So who needs that much capacity and operating speed? Primarily, video professionals who work with 4/5/6K cameras. According to LaCie, “That kind of speed can slash time off nearly every post-production workflow task. Users can ingest hours of RAW footage from a Blackmagic® cinema camera into Adobe® Premiere® Pro in a fraction of the time. They can then edit multiple streams of ProRes 422 (HQ), ProRes 4444 XQ, as well as uncompressed HD 10-bit and 12-bit video. For big projects packed with high-def clips, thumbnail and preview rendering becomes much more responsive.”

No price had been announced as of this writing but, for comparison, the 12 TB 2big Thunderbolt 2 unit goes for \$899, so it won’t be cheap. ▲

■ LaCie’s 12big Thunderbolt 3 storage unit holds up to 96TB.



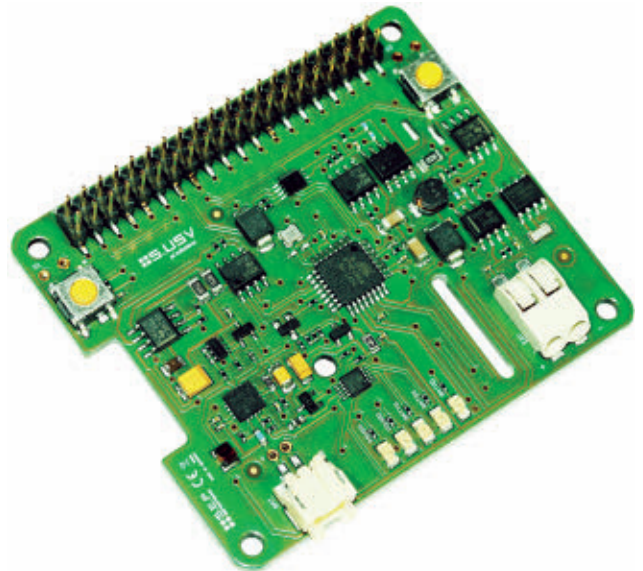
CIRCUITS and DEVICES

Supply for Your Pi

It’s hard to guess how many Raspberry Pi computer applications are critical enough to require an uninterruptible power supply (UPS), but a German company, Seprotronic GmbH, has developed two models of them: the “S.USV pi basic” and the “S.USV pi advanced.” The basic model — described as a fully functional plug and play solution — offers the ability to adjust the integrated charging current to 300 mA (standard), 500 mA, or 1A, allowing a much shorter battery charging time.

As it is connected directly through the J8 connector on the Raspberry Pi, it uses a common voltage source, so no additional cabling or power supply is needed. The module is equipped with a lithium polymer battery, and an integrated boost switching power converter covers the necessary voltage range. The “advanced” version also provides a power input for the extended voltage range of 7V to 24V (for solar cells, automotive applications, etc.).

Pricing is €29.99 (about \$34 at the current exchange rate) for the basic unit and €54.99 (\$62.23) for the advanced. Details are available at www.s-usv.de/index_en.php. ▲



■ The S.USV pi basic, a UPS for all Raspberry Pi models.

CIRCUITS and DEVICES Continued

Never Lose Anything Again

You may as well admit it. You had this idea years ago, but just never came up with a way to implement it. Well, it's time to kick yourself, because the Chipolo wireless object locator is here and available at chipolo.net. You just attach a Chipolo disc to practically anything (keys, dog, wallet, crazy Uncle Chester, etc.) and download the app to your smartphone. As long as the item is within Chipolo's Bluetooth range (200 ft, or 61 m), you can make it play a loud tune.



■ Chipolo keeps track of keys or anything else.

If you can't find your phone, just shake one of your Chipolos, and the phone will ring. If the phone is too far away, you can log onto the company's website app to see its last known location, make it ring, or send a lock-screen message to anyone who has found or stolen it. The discs come in your choice of nine colors, which can be coded to the attached item. You may as well stop kicking yourself and buy a few. The price is \$29.95 for a solo unit, \$49.95 for a pair, or \$89.95 for a family four-pack. Each comes with an extra battery, which is said to last six months. ▲



INDUSTRY and the PROFESSION

What Happened to Bell Labs?

Maybe you have had occasion to wonder what ever happened to AT&T Bell Labs, the famous R&D company whose roots go back 91 years to Alexander himself. After all, Bell Labs researchers have been awarded eight Nobel prizes and brought us things like lasers, radio astronomy, the transistor, Unix, and so on. You might also have wondered what happened to Nokia Corp. — the Finnish communications company that conned Microsoft into paying \$7.7 billion for a dying line of smartphones (which continues to decline in market share).

As it turns out, when Nokia acquired Alcatel-Lucent last year, Bell was part of the deal, and it is now known as Nokia Bell Labs (www.bell-labs.com). In a recent interview, Bell president, Marcus Weldon expressed satisfaction with the merger and confidence that the lab's traditional approach of looking at projects "from the future, back" will continue. This refers to the concept of looking a decade ahead and imagining how to get there from here rather than just moving forward from wherever you are.

According to Weldon, if Bell receives enough funding to maintain its traditions of innovation and integrity, the future will remain very bright. He also noted that if that doesn't happen, the clearest sign will be that "I will leave." Time will tell. **NV**



■ Bell Labs founder, none other than Alexander Graham Bell himself.

In this column, Tim answers questions about all aspects of electronics, including computer hardware, software, circuits, electronic theory, troubleshooting, and anything else of interest to the hobbyist. Feel free to participate with your questions, comments, or suggestions.

Send all questions and comments to: Q&A@nutsvolts.com.

Enlightenment on Modern Lighting

Q I see a lot of ads for LED and CFL lighting advertised as a replacement for incandescent lights. Which is better?

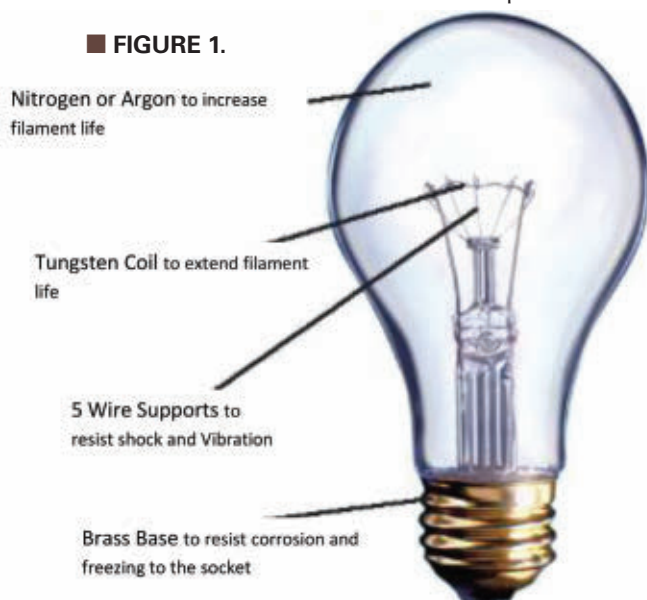
Joshua Peebles
Battle Creek, MI

A Compact Fluorescent Lamps (CFL) and Light Emitting Diode (LED) lamps were designed as a low energy consuming alternative to residential standard base screw-in incandescent lamps.

Figure 1 shows the internal components of an incandescent lamp. The incandescent lamp consists of a Tungsten filament enclosed in a glass bulb which is filled with inert nitrogen or argon gases. When the filament conducts electricity, it heats to approximately 3,100 to 5,500 degrees Fahrenheit, where it glows and converts approximately two percent of the electrical power input into visible light and 98 percent into heat with a trace of ultraviolet (UV) light.

Incandescent lamps are very efficient generators of heat (a.k.a., great wasters of electricity), but not too efficient as light emitters. [Our instrument shop used several 100 watt incandescent bulbs wired in parallel as a

■ FIGURE 1.



● Enlightenment on Modern Lighting

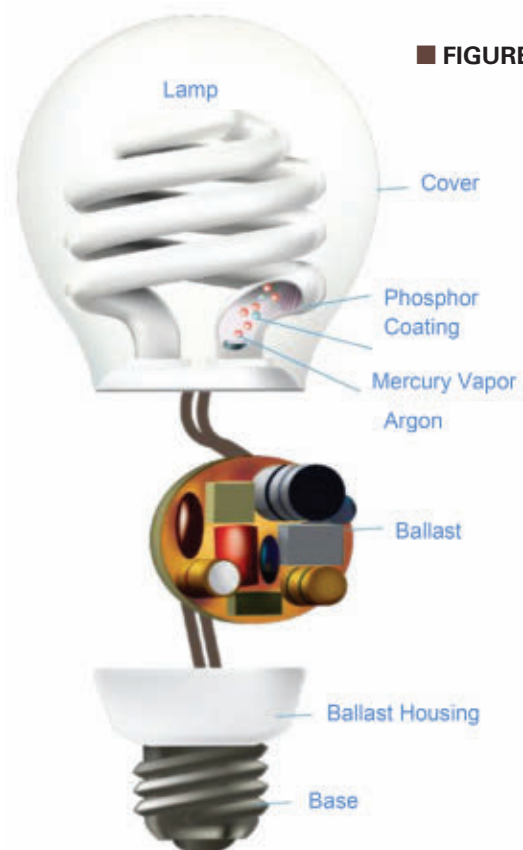
● Is "Green Energy" Really Green?

dummy load for testing DC motor controllers.] A 60 watt incandescent lamp radiates around 800 lumens of visible light, which is approximately 13.33 lumens per watt of input electrical power. A 60 watt incandescent lamp costs approximately \$0.50 and will last approximately 1,200 hours.

Figure 2 shows the internal construction of a CFL. The CFL is essentially a tube type fluorescent lamp twisted into a helical package. As such, it requires a ballast to produce a sufficient voltage to cause the lamp to fluoresce and to regulate current to the lamp after the fluorescence starts.

The process of fluorescence uses mercury (3 to 5 milligrams per lamp of an extremely toxic substance) vapor in argon gas. When an electrical voltage is placed across the lamp, the mercury atoms ionize and when the electrons recombine with the atom, UV light is emitted. This UV light energizes a phosphor coating on the inside of the CFL's glass tube which emits visible light. My experience with CFLs is that they sometimes take a while to reach the full light output and proper color — especially in cold ambient conditions — and they sometimes generate some heat.

■ FIGURE 2.



QUESTIONS and ANSWERS

Post comments on this article at
www.nutsvolts.com/magazine/article/september2016-QA

A CFL with a 14 watt electrical power input emits approximately 800 lumens (similar to a 60W incandescent) of light, which is approximately 57 lumens per watt of input electrical power. A 14 watt CFL costs approximately \$1 and will last approximately 8,000 hours.

Figure 3 shows the construction of an LED lamp which uses a number of individual LEDs wired together to produce the desired amount of light. An LED with a six watt electrical power input emits approximately 800 lumens (similar to a 60W incandescent) of light, which is approximately 133 lumens per watt of input electrical power. A six watt LED lamp costs around \$10 and will last approximately 25,000 hours. With improvements in manufacturing and an increase in consumer purchases, the costs of LED lamps should drop in the future.

Table 1 gives a cost comparison of incandescent, CFL, and LED lamps based on using the 800 lumen light output lamp six hours per day, and an electrical power cost of \$0.12 per Kilo-Watt-Hour (12 cents per KWH).

Based on the life of 11.4 years for an LED lamp, the operating costs of the incandescent lamp are 6.84 times that of an LED, and 4.25 times that of a CFL. The operating costs of the CFL are 0.23 times that of an incandescent and 1.60 times that of an LED. The operating costs of the LED lamp are 0.15 times that of an incandescent and 0.62 times that of a CFL. Clearly, the LED is superior in the long run.

My recommendation is to replace the existing bulbs a few at a time so you don't incur a tremendous drain on your pocketbook. I used this approach to replace incandescent bulbs with CFLs which save me a tremendous drop in electrical costs. My calculations have convinced me to start replacing the CFLs with LEDs in the near future.

Besides saving money and work replacing lamps for

myself, I will save the power company's capacity which will lower their operating costs. Plus, I will save carbon emissions to the atmosphere which everyone agrees will help regardless of your stance on climate change. Replacing the CFLs will also reduce the mercury hazard that may leach into the environment.

I hope this short analysis helps you to decide which type of electrical lamps you need to use. It sure has helped me.

Tim Brown

Is "Green Energy" Really Green?

Q I see a lot on TV and in magazines about green energy that is supposed to be environmentally friendly, but are these devices really as benign as they are purported to be?

**Lynette Adams
Hardwick, VT**

A This is both an emotionally and politically charged question. I will try to stay as neutral as possible with my answer, but if I hit one of your hot buttons let me know and we can discuss this further.

Green energy is also known as renewable energy which can be replaced or regenerated on a short time frame as opposed to the span required to replenish fossil fuels (coal, crude oil, and natural gas). I will stick to the green energy sources that are used to generate electricity since there are a number of papers about green energy sources for heating, cooling, and transportation for those who are interested.

Green energy sources include solar photovoltaic, wind, tidal/wave, hydroelectric, geothermal, biofuel, biomass, and solar thermal. I will divide these green energy sources into thermal (geothermal, biofuel, biomass, solar thermal) and non-thermal processes (solar photovoltaic, wind, tidal/wave, hydroelectric) because of similar characteristics.

First, let's look at fossil fuels and some potential reasons to not use them for our long term energy needs. According to scientists, fossil fuels are derived from carbon bearing materials that were buried millions of years ago. The combination of heat and pressure converted to hydrocarbon gases, liquids, and solids known to us in modern times as natural gas, crude oil, and coal, respectively. Natural gas and coal can be burned to generate heat (thermal) energy, but crude oil must be refined to produce gasoline, kerosene, diesel, and fuel oil which are burned to create thermal energy.

These hydrocarbon fuels burn in the presence of oxygen (chemists call this oxidation) to create water (H_2O), carbon dioxide (CO_2), and heat. This combustion process occurs according to the chemical equation for burning

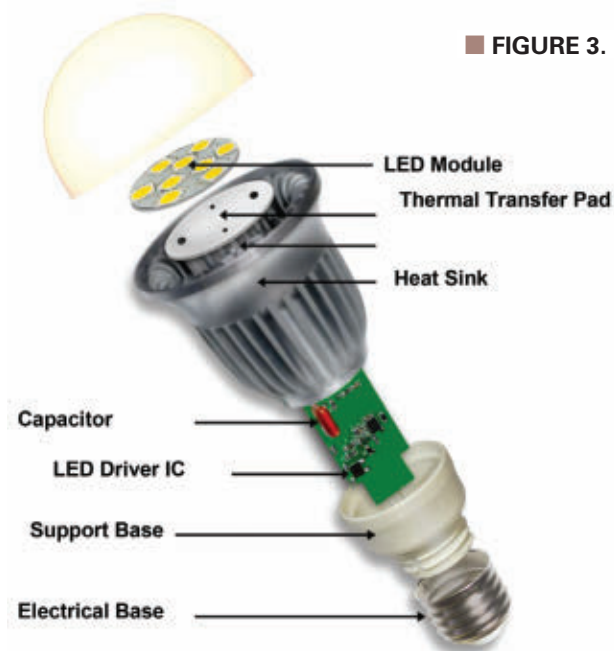
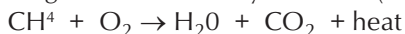


Figure Credits

1. <http://conservationteam.com/incandescents.html>
2. www.energystar.gov/sites/default/files/assets/images/Parts_of_CFL_large.jpg
3. http://m.eet.com/media/1188517/0513pmdl_iwatt01.jpg

natural gas which is mostly methane (CH_4):



Water generated by fossil fuel combustion is not a problem since all known life forms need water in some form to survive. Carbon dioxide, however, has been implicated by environmentalists and climate scientists as a major portion of the greenhouse gases which are thought to trap solar thermal energy and thus cause the Earth's atmospheric temperature to rise.

Earth's atmosphere has positive feedback processes such as increased release of CO_2 from the oceans with increasing temperatures, but there are also negative feedback mechanisms such as evaporation of water which increases cloud cover, which then blocks solar irradiation and thus reduces surface temperatures. NASA estimates that water vapor is from 60 to 70 percent of the greenhouse effect which means the anthropogenic (manmade) proportions of CO_2 are about 0.2 to 0.3 percent of the total CO_2 generated annually.

According to the American Chemical Society, there are other greenhouse gases such as methane, nitrous oxide, ozone, and halogenated gases (refrigerants, insulators, and cleaning agents), but water vapor and CO_2 are the major causative agents of the greenhouse effect.

When you hear the term "carbon footprint," you are basically hearing the amount of carbon dioxide generated by a human activity. So, people try to find energy sources which generate little or no carbon dioxide ("green energy"). Let's look at these alternative energy sources and their environmental effects.

THERMAL PROCESSES

Solar Thermal

Solar thermal energy is spoken of as "free energy" since the Sun is always radiating our planet with approximately one kilowatt of thermal energy per square

meter at sea level on a clear day at local noon (called insolation). The free part goes away when we start trying to capture this solar energy.

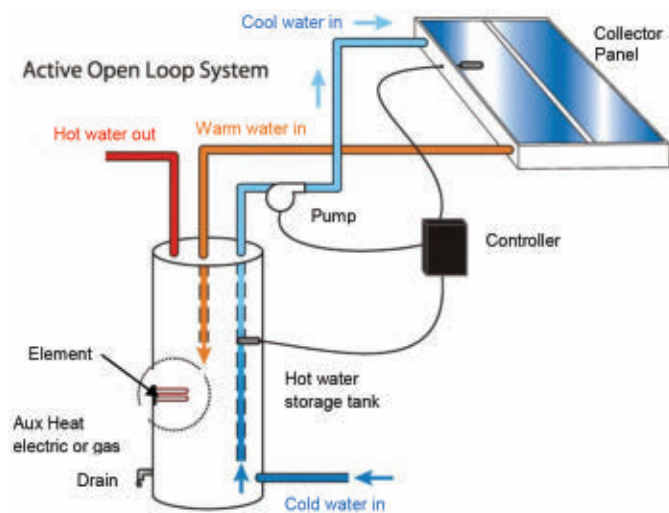
Figure 4 shows a simplified solar thermal collection system for providing residential hot water. This same type of system with the addition of a parabolic concentrating mirror could also be used to heat a fluid which could, in turn, drive an electric generator. Having solar collectors track the varying position of the Sun improves the output of the system, but requires additional energy and complexity.

The carbon footprint of the solar thermal system during operation is zero. The additional environmental impact would be the CO_2 and waste products created during the manufacture of materials to build the system such as plastic, steel, and copper, plus the energy needed to create these materials. A downside of the solar thermal systems is the amount of sunshine available, which is lowest during the time we need it the most if we are heating our buildings (in the winter and further north). If simply generating electricity, the demand is hopefully much less during the winter.

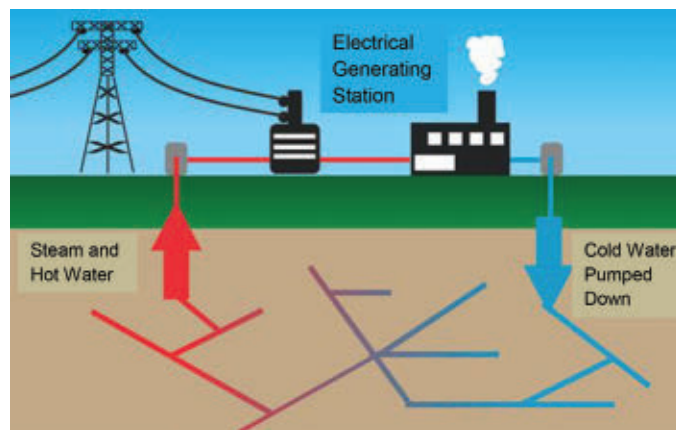
Geothermal

Geothermal energy is present in the heat available in the soil (heat pump), hot water (geysers and hot springs which are very regional sources), and in volcanically heated rocks (also highly regional). **Figure 5** shows a geothermal electrical generating station which pumps cold water down into hot rocks deep underground and the steam returned is used to drive a generator.

The environmental impact would be the CO_2 and waste products created during the manufacture of materials to build the system such as concrete and steel, plus the energy needed to create these materials and construct the station. The downsides are possible corrosive and toxic materials returned from underground, and the possibility of earthquakes from pumping water into the hot rocks similar to those caused by hydraulic fracturing (fracking) in the petroleum industry.



■ **FIGURE 4.** Open loop active solar energy residential heating system.



■ **FIGURE 5.** Geothermal electrical generation station.

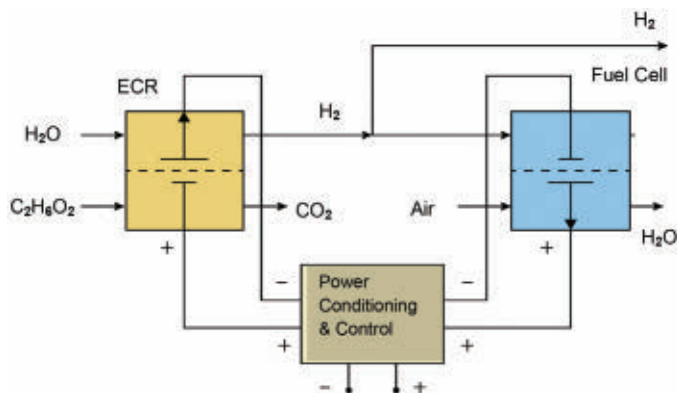
Biofuel

Biofuel is any fuel produced by a biological process such as agriculture, fermentation, or the harvest of plants that produce oil-like substances. We are familiar with the ethanol fuels made by fermenting corn (some people have tried this at home and ran afoul of the law, so don't try this without proper licensing) and the biodiesel extracted from soy beans. Algae and fungi have been used to produce liquid fuels.

Years ago, I read an article about a dairy farm that slurried cow manure in water, placed the mixture into a huge air-tight rubberized vessel, and extracted the methane gas produced by the bacteria digesting the slurry to run farm equipment and provide heat.

Any plant material can be used as a biofuel or processed to produce a biofuel. Some problems with plant based renewable fuels are food crops such as corn will not be available for humans or animal feed which could cause food chain problems; the farming activities used to grow crops for biofuels often consume more energy than fuels produce; biofuels are often incompatible with current engines (for example, ethanol is not permitted on piston aircraft engines); and the carbon footprint of the biofuel process is very large since agricultural runoffs often lead to the production of CO₂.

Biofuels can be burned in various types of boilers to generate steam to drive an electrical generator. The construction of agricultural machinery also requires energy and produces pollutants, plus the transportation of the biofuel precursors has a carbon footprint. Biofuels do remove some of the CO₂ as a part of growing the plants, but overall there is a positive carbon footprint.



■ **FIGURE 6. Biomass electrical generating system using an Electrocatalytic Reformer (ECR) and fuel cell.**

C₂H₆O₂ - Ethylene Glycol

H₂ - Hydrogen

CO₂ - Carbon Dioxide

H₂O - Water

Air - 79 percent nitrogen (N₂) and 21 percent oxygen (O₂)

The ECR chemical reactions are:

Anode: C₂H₆O₂ + 2 H₂O → 2 CO₂ + 10 H⁺ + 10 e⁻

Cathode: 2 H⁺ + 2 e⁻ → H₂

The fuel cell chemical reactions are:

Anode: 2 H₂ → 4 H⁺ + 4 e⁻

Cathode: 4 H⁺ + O₂ + 4 e⁻ → H₂O

Biomass

Biomass is organic material obtained from living (or recently living) plants. Biomass is usually a waste product of another process such as the wood chips from the pulp and paper process. The burning of waste wood or agricultural wastes to produce useful heat has been employed for many years. Biomass is often converted to a biofuel which can be burned to generate heat, such as the fermentation of sugar beet wastes to produce ethanol.

A way of producing electricity directly from biomass is the fuel cell. In the biomass fuel cell, the biogas or biofuel produced from the biomass is used in a fuel cell as shown in **Figure 6**.

The Electrocatalytic Reformer (ECR) uses a platinum anode to break the ethylene glycol into CO₂ and hydrogen ions. At the ECR's platinum cathode, the hydrogen ions combine in pairs to produce H₂ molecules. The H₂ molecules pass through a proton exchange membrane and are sent to the fuel cell while the CO₂ molecules are exhausted to the atmosphere.

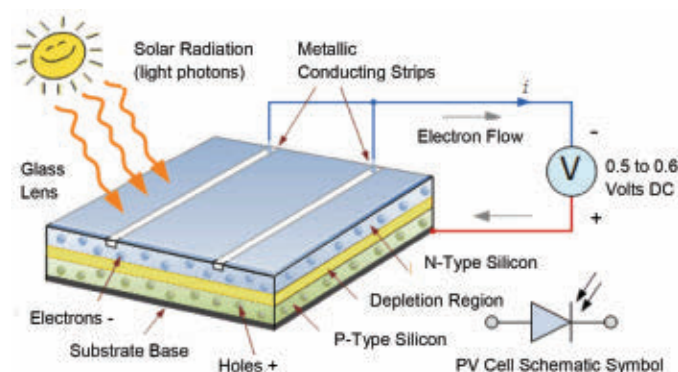
This example uses ethylene glycol for simplicity, but a commercial process would use five-carbon and six-carbon sugars obtained from the waste streams of another process. The H₂ molecules at the fuel cell's platinum anode break up into hydrogen ions and electrons which are sent through an electrical circuit to perform useful work. At the fuel cell's cathode, these electrons combine with the hydrogen ions and oxygen from the air pumped into the fuel cell to create water (H₂O) which can then be used for other processes such as drinking water on the Space Shuttles.

The emission of carbon dioxide from the biomass ECR/fuel cell system does create a carbon footprint for the process, but it is less than most fossil fuel systems.

NON-THERMAL PROCESSES

Solar Photovoltaic

Solar photovoltaic energy systems use solar cells like the one shown in **Figure 7** to produce electron current flow from sunlight. The solar cell is a P-N diode that has



■ **FIGURE 7. Principle of operation for a solar photovoltaic cell.**

Figure Credits

7. www.alternative-energy-tutorials.com/solar-power/photovoltaics.html

8. <http://solarenergy-usa.com/product/solar-pv-panels/>

9. www.power-technology.com/features/feature-biggest-wind-farms-in-the-world-texas/

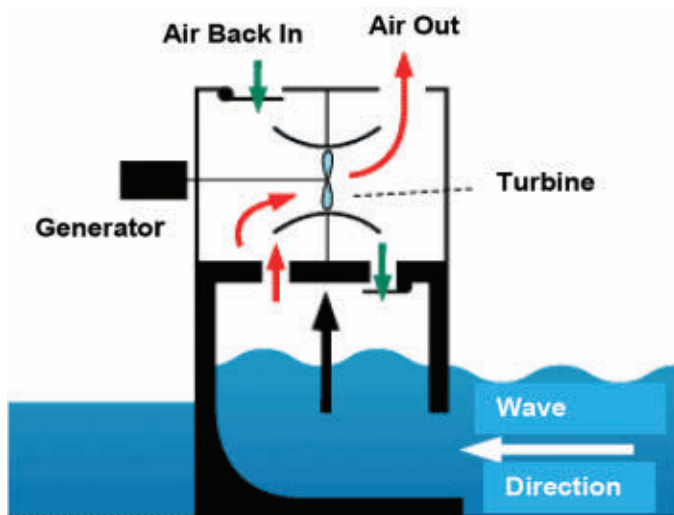


■ FIGURE 8. Solar photovoltaic panels.

an opening to allow sunlight to focus on the P-N junction where the incident photon energy creates hole-electron pairs. The electrons migrate into the N-material creating holes in the P-material which causes an electrically neutral zone called the depletion layer or region. The separation of charges across the depletion layer creates a voltage of 0.5 to 0.6 VDC. If the solar cell's metallic contacts are connected to a load, a small current flows (approximately three amps at 0.5 VDC with full insolation of 1,000 watts/square meter).

Solar cells produce a fair amount of current, but not enough voltage to be useful. So, many individual cells are connected together in series banks to increase the overall voltage output, and a number of banks are connected in parallel modules to increase the overall current output. A number of these solar modules are connected in series-parallel arrays to further increase the volt and current output.

The solar arrays are enclosed in metal frames with glass covers for protection from the environment's elements



■ FIGURE 10. Wave energy electrical generation system.



■ FIGURE 9. Wind power farm at the Alta Wind Energy Centre (AWEC) in Tehachapi (Kern County), CA which generates 1,550 megawatts of electrical power.

(rain, sand, etc.), and are called solar panels. **Figure 8** shows a 4 x 3 solar panel where you can see the individual solar cells as the small squares and the arrays with a 6 x 10 arrangement of 4 inch x 4 inch solar cells. Solar panels seem like the way to go for protecting the environment but if we look into the solar cell manufacturing processes, we see the use of corrosive materials such as sodium hydroxide and hydrofluoric acid. Plus, some green house gases are generated, and the necessary energy to melt sand to make the solar cell wafers comes into play. However, the overall effect of using solar photovoltaic energy has a smaller carbon footprint than that of fossil fuel powered systems.

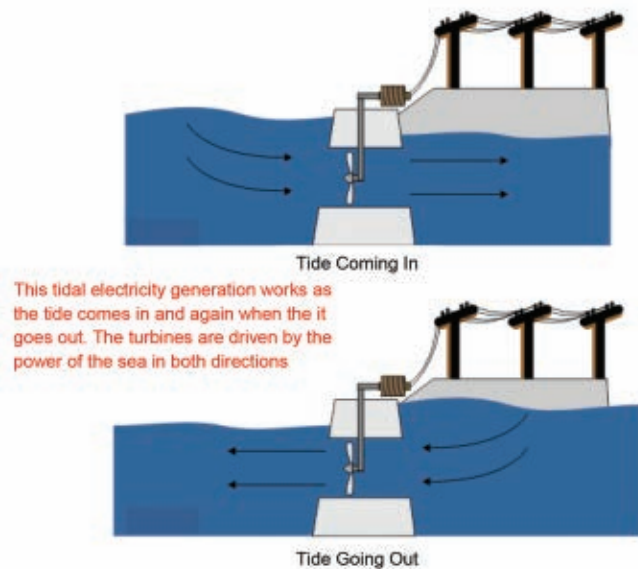
Wind

This type of generated energy uses wind blowing through wind turbines which rotate and turn electrical generators. To make wind energy economically feasible, you have to have a consistent wind velocity of 10 to 13 miles per hour. Wind turbines are installed as high as possible above the local terrain to maximize the wind speed. Additionally, the wind turbines need a mechanism to "feather" the turbine blades during periods of high winds to prevent damage. The turbines are also notorious for killing raptor birds.

Figure 9 shows a number of wind turbine electrical generators arranged over several acres to form a wind farm. Most wind farms are located in the Plains States and Rocky Mountain States in the US. The only carbon footprint for wind energy comes from constructing the wind turbines, towers, and generation/storage systems. Wind energy can be scaled to the point that you can build your own roof top system, but be aware that the turbines can be fairly noisy.

Tidal/Wave

Tidal and wave energy use the ebb and flow of the oceans to generate electricity. Wave energy uses the short

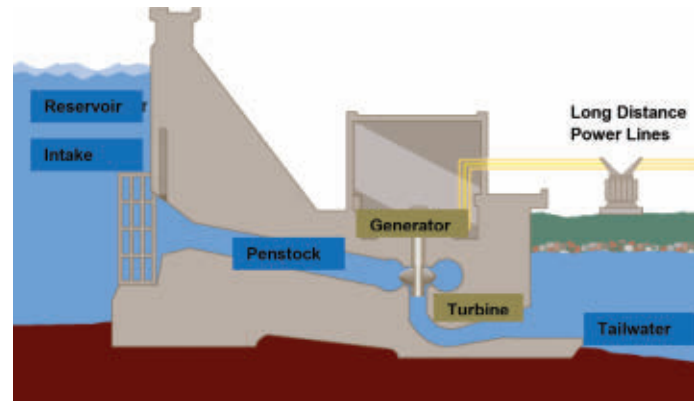


■ **FIGURE 11. Tidal energy electrical generation system.**

time frame movement of ocean waves, while tidal energy uses the long time frame motion of the ocean tides. **Figure 10** shows a wave power generation system and **Figure 11** shows a tidal power generation system. Other than the usual issues from building the wave and tidal power plants, there is no carbon footprint. However, these facilities may interfere with coastal activities such as shipping and recreation. Obviously, these must be located in coastal areas, so they are not available to most areas of the US.

Hydroelectric

Hydroelectric energy is derived from large quantities of water from a dammed up lake or reservoir flowing through turbines that are used to drive an electric generator. **Figure 12** shows a typical hydroelectric power plant. The key to generating power with water is to have the highest amount of water above the level of the generators (head) as possible.



■ **FIGURE 12. Hydroelectric electrical generation system.**

This electrical power is found using the formula $P = 0.63HQe$ where P = kilowatts of electricity generated, H = number of feet of water above the generator, Q = gallons per minute of water flowing through the generator, and e is the overall efficiency of the turbine and generator system (approximately 0.5).

To generate 10 KW of electrical power with a 10 foot head of water would require around three gallons per minute of water (a 100 foot long 1/2 inch garden hose maximum flow is approximately 6 gpm). An average home in the US uses 10 Kwh per day, so this 10 KW generator would have to run 24 hours a day/seven days a week. This would be 4,320 gallons of water per day, or 129,600 gallons per month (last month, my house used 2,600 gallons of city water for two people) which is a lot of water, but the 10 foot head is beyond most hobbyist's level (the creek along my boundary line may have a one foot head).

See Q&A Sidelines for a comparison and more information on each of these energy types.

I hope I have answered your question regarding renewable energy sources.

Tim Brown
N&V Q&A

Q&A SIDELINES

Is "Green Energy" Really Green?

Comparison of Renewable Energy Sources

www.renewable-energysources.com/
www.renewableenergyworld.com/articles/print/rewna/volume-1/issue-1/solar-energy/how-to-compare-power-generation-choices.html

Solar Thermal Energy

www.eia.gov/energyexplained/?page=solar_thermal_power_plants

Geothermal Energy

www.eia.gov/energyexplained/index.cfm?page=geothermal_home

Biofuel Energy

www.eia.gov/energyexplained/index.cfm?page=biofuel_home

Biomass Energy

www.eia.gov/energyexplained/index.cfm?page=biomass_home

Solar Photovoltaic Systems

www1.eere.energy.gov/buildings/residential/pdfs/rerh_pv_guide.pdf

Wind Energy

www.eia.gov/energyexplained/index.cfm?page=wind_home

Tidal/Wave Energy

www.oceanenergycouncil.com/ocean-energy/tidal-energy/
www.oceanenergycouncil.com/ocean-energy/wave-energy/

Hydroelectric Energy

www.eia.gov/energyexplained/index.cfm?page=hydropower_home

Common Antennas You Need to Know About

Everything acts like an antenna — sometimes, the antenna is you!

Are you reading this on a tablet? The tablet depends on its built-in antenna for Wi-Fi connectivity, transferring files back and forth from another PC, or to use one of the many mobile phone networks blanketing populated areas. Even in the miniature flattened volumes that are the norm for today's wireless communication equipment, there has to be an antenna. It might be hidden, it might be tiny, but it's there. Knowing about common antennas is a good step toward effective data link (and other wireless) system design.

Dipole Basics

The very first installment of this column showed how to build a simple ground-plane antenna from a BNC connector and some lengths of wire. The ground-plane is basically "half a half-wave dipole," so let's back up a little bit and review the dipole. As the fundamental element at the root of many antenna designs, understanding the dipole places you on solid ground.

When referring to a dipole, the usual assumption is that the dipole is a thin conductor (like wire or rod), one-half wavelength ($1/2 \lambda$) long at the frequency being used, and that the feed line is attached in the center (a.k.a., "center-fed"). In fact, "dipole" is constructed from "di" (two) and "pole" (polarities) which describe the signal's effect on the dipole: The polarity of the voltage between the ends of the dipole reverses every half-cycle, causing current flow to reverse as well, sloshing back and forth along the dipole. Voltage is maximum at the ends of the dipole, and current is maximum in the middle as illustrated by **Figure 1**.

The dipole radiates a signal caused by the electrons

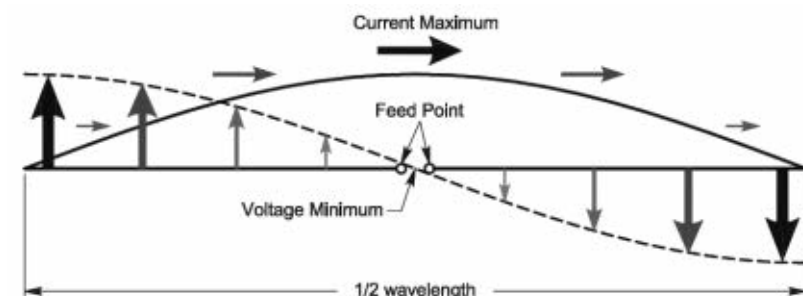


FIGURE 1. Current and voltage distribution along a half-wave dipole.
(Graphic courtesy of the American Radio Relay League.)

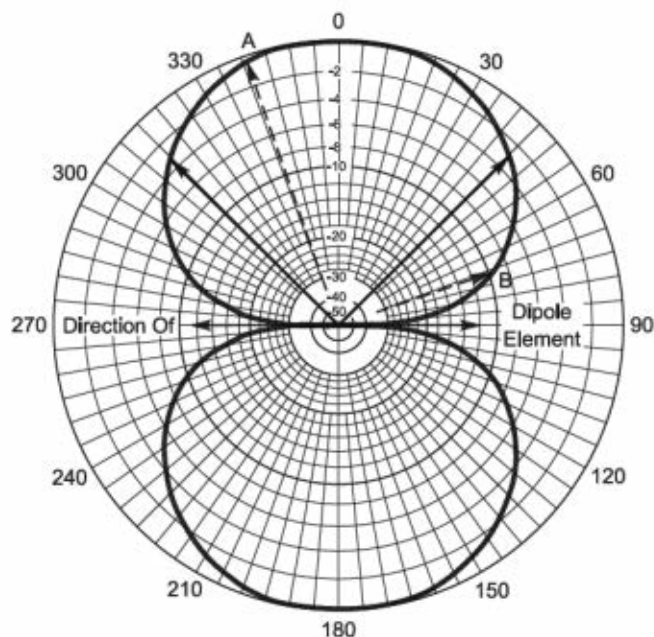


FIGURE 2. Radiation pattern of a half-wave dipole in free space shows the strongest signal radiated and received broadside to the dipole's axis. Distance from the origin to the solid curve represents signal strength in dB with respect to the maximum. The solid arrows at 45 and 135 degrees show the points at which signal strength is reduced by half (-3 dB).
(Graphic courtesy of the American Radio Relay League.)

that make up the current being constantly accelerated or decelerated in response to the AC voltage created by energy from the feed line. In fact, radiation of electromagnetic (EM) waves only occurs when electric

charge experiences positive or negative acceleration. (In radio and electronics, the charges are electrons but they could be anything with an electric charge.) The strongest radiation of EM waves results from charge being accelerated along a straight line (like the dipole conductor); it is strongest at right angles to the flow of the charge. This results in the classic "figure 8" pattern of the dipole's radiation as shown in **Figure 2**.

A physical dipole is a little shorter than the half-wavelength of an EM wave traveling in "free

Post comments on this article and find any associated files and/or downloads at www.nutsvolts.com/magazine/article/September2016_Ham_Workbench_Common-Antennas.

Antennas Work Coming and Going

Regarding the radiation pattern in **Figure 2** — does it describe the antenna's transmitting characteristics or receiving? It describes both!

Nearly all antennas are bilateral, meaning they behave the same for transmitted or received signals. This ability requires the antenna to not contain any non-linear materials such as some ferrites or magnetic materials. The antenna must also be operated at power levels that do not cause the antenna materials to operate non-linearly, such as from arcing or saturating.

Radiation?! Eek!

The word "radiation" tends to freak people out. In wireless, the radiated energy is non-ionizing and unable to liberate an electron from its host atom which would lead to chemical changes and, eventually, genetic damage. Electromagnetic radiation has not been shown to have any effect on living tissue except to heat it up — called "thermal effects."

The term "radiation" is just used as a generic term for "giving off energy." The frequency of the EM waves would have to be very, very, very much higher than radio frequencies — even microwave — before it would be of concern.

Ultra-violet rays that cause sunburn are at about the lowest frequency (longest wavelength) for which genetic damage would occur, and thus present a cancer risk. Wireless communication waves are many orders of magnitude less energetic and thus pose no such threat.

space," meaning away from any conductive or refractive material. This is due to the effect of the wave traveling along the surface of the dipole's conductor, called the antenna element. As the element is made thicker (larger wire, thicker rod, wider strip, etc.), the shortening effect gets stronger. You may see the equation length (in feet) = $468 / \text{frequency (in MHz)}$, but that is only applicable at relatively low frequencies (below 20 MHz), and for thin wire dipoles within a half-wavelength of the ground.

If you are building a dipole, determine the element dimensions by using an antenna modeling program that can account for this effect, use published lengths for a proven design, or be prepared to test and trim until you get the results you want.

Yagi Antennas

Take a look above public safety agency stations, weather stations and tide gauges, government buildings, and so forth, and you'll see small antennas with several parallel elements made from aluminum rod. Wi-Fi and mobile phone "extender" antennas look the same as well, with numerous short parallel elements arranged along a central support (called a boom). These are Yagi (pronounced to rhyme with "baggy") antennas; more properly called Yagi-Uda arrays after the antenna's two inventors, Drs Yagi and Uda from the University of Tokyo. Developed in the mid 1920s, the Yagi antenna has been a mainstay of HF, VHF, UHF, and microwave wireless

What is Antenna Gain?

When a designer says a beam antenna has a gain of some number of dB, what do they mean?

First, the antenna does not create energy, it merely focuses it. Like a parabolic mirror, it does not generate light; it only reflects it in such a way as to concentrate it. Gain antennas do the same.

Imagine the isotropic antenna that radiates and receives equally in all directions. Its radiation pattern is a perfect circle. This antenna does not exist in practice, but is a useful mathematical reference.

Imagine the isotropic antenna's three-dimensional pattern as a round balloon with the air representing the radiated signal. A directional antenna "squeezes the balloon" so that the signal is suppressed in some directions and enhanced in others. It does not "create" energy — it merely redistributes it.

The comparison between the directional and isotropic antennas is measured in dBi — decibels with respect to an isotropic antenna which forms the reference for the calculation of dB. An antenna with 3 dBi of gain will transmit or receive a signal 3 dB stronger than an isotropic antenna in that specific direction.

Another common reference antenna is the dipole, and gain (with respect to a dipole) is measured in dBd with the reference being the dipole's maximum free space gain broadside to the dipole as shown in **Figure 2**.

communication ever since. (While Dr. Uda ["oo-dah"] was the primary inventor, his partner, Dr. Yagi could speak English and so became more closely identified with the antenna during a series of lectures introducing the design.)

The Yagi is a parasitic array, meaning that only one of the elements (the driven element) is actually connected to a feed line and the rest interact with the radiated EM waves. Each element is very similar to a dipole, but if you look closely, you'll see that the elements get longer toward the "back" of the antenna (the non-preferred direction) and shorter toward the "front." The driven element is usually just a few percent shorter than an independent half-wavelength dipole. Elements in back of the driven element are called reflectors, and those in front of the driven element are called directors.

The non-driven elements interact with the signal radiated by the driven element — an effect called mutual coupling. Coupling means that the elements exchange energy by picking up and re-radiating signals. The re-radiated signal is also picked up and re-radiated by the other elements again and again. The coupling between elements, the spacing between them, and the effect of the element lengths on current in elements combine to reinforce signals toward (or from) the front and reject signals toward (or from) the back. This creates a "beam" antenna with a preferred direction for receiving and transmitting, which is called "gain" as explained in the sidebar.

Figure 3 shows a three-element Yagi used for portable operation in direction finding. (For more information on these popular activities, see

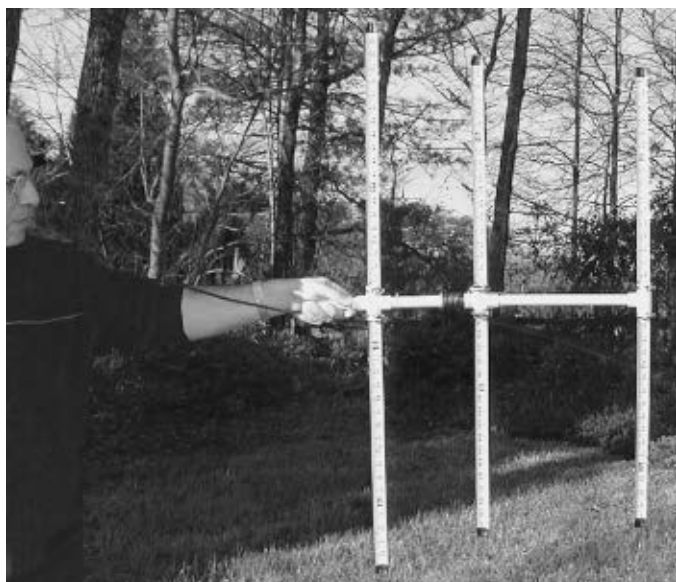


FIGURE 3. A three-element Yagi antenna made for portable direction finding use. The antenna consists of a reflector (left), driven element (middle), and director (right). This antenna has approximately 5 dB of gain in the forward direction (to the right) compared to the maximum signal radiated by a dipole. (Graphic courtesy of the American Radio Relay League.)

The Biggest Yagi

The largest single amateur radio antenna ever built was a three-element behemoth Yagi for the 160 meter band (1.8 MHz) by the Finnish club, Radio Arcala, OH8X (vk6ysf.com/Radio_Extremes.htm). It had elements 59 meters long atop a 100 meter tower, and a boom so large a small car could be placed inside of it.

Compare that to a Yagi for Wi-Fi with the longest element approximately 6 cm long! Yet, the antennas operate on exactly the same principle of re-radiation and reinforcement.

Hams have a saying that, "If it stayed up last winter, it's not big enough!" Well, the OH8X Yagi was certainly big enough, and unfortunately validated the saying by collapsing in the winter of 2013. Ironically, it was not the antenna that failed, it was the supporting tower!

[en.wikipedia.org/wiki/Amateur_radio_direction_finding.\)](http://en.wikipedia.org/wiki/Amateur_radio_direction_finding.)

This particular antenna's elements are made from metal measuring tape so they can flex and bend without damage. You can clearly see the reflector at the back, the driven element in the middle, and the director element at the right. **Figure 4** shows the radiation pattern typical for such an antenna. Yagi antennas with a dozen elements or more — creating a very narrow beam — are not uncommon for frequencies above 100 MHz. At lower frequencies, the longer elements and boom require more robust construction materials and techniques.

If you would like to experiment with a Yagi antenna to improve your TV reception, pull in a distant FM station, or maybe extend the range of a 900 MHz data link, you can do so inexpensively by building the antenna yourself. Kent Britain WA5VJB has developed an entire line of Yagi



FIGURE 4. The T-28 log-periodic by Tennadyne covers 50 to 1,300 MHz. The 12 foot boom consists of a pair of square tubes that act as the feed line for the antenna. (Photo by the author.)

antennas that can be constructed from nothing more than spare copper wire and a piece of 1x2 lumber. Kent's "Cheap Yagis" website (www.wa5vjb.com/yagi-pdf/cheapyagi.pdf) provides all the details.

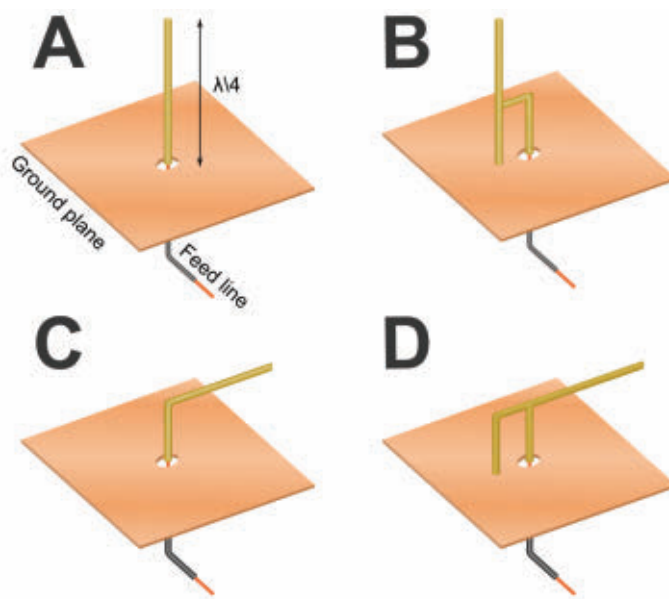
Log-Periodics

You may be thinking, "Ah hah, so my outside TV-FM antenna is a Yagi!" Not quite. While common TV antennas look like a Yagi, closer inspection shows that the dipole elements are bent forward in shallow vees and the feed line is connected to every element in a criss-cross pattern down the length of the array. This type of antenna is a log-periodic dipole array (LPDA, or more commonly, a log-periodic or just plain "log." **Figure 4** shows a log-periodic antenna for 50 MHz through 1,300 MHz mounted above my house (the antenna elements are all horizontal — the antenna is not pointing at the sky).

The log-periodic's name comes from the logarithmic spacing and length of the elements; it is defined entirely in terms of angles, such as the taper of the triangle surrounding the elements. The active region of the antenna "moves" back and forth along the array as the frequency changes. The dipole elements closest to the frequency of the signal radiate it or transmit it while the other elements remain electrically inert.

With a sufficient number of elements, a log can cover a very wide range of frequencies with consistent performance — even on the shortwave HF (high

FIGURE 5. The Inverted-F antenna is a modified ground-plane (A) with the feed line connected at a point with the desired impedance (B). If the antenna is bent over the ground-plane at right angles (C), the feed point can still be attached away from the base (D) to provide a good impedance match. (Graphic courtesy of Spinningspark at Wikipedia.)



More Antenna Resources

The ARRL (American Radio Relay League) offers a lot of information on antennas to the public via the Technical Information Service (www.arrl.org/radio-technology-topics) and on the Tech Portal page (www.arrl.org/tech-portal) with links to third-party vendors and amateur reference sites. We've just scratched the surface of the world of antennas — these sites will help you learn more.

frequency) bands from 3-30 MHz. In fact, these antennas are popular with government and military stations because only one large antenna is needed to operate effectively on several bands instead of several dedicated antennas.

Inverted-F

One of the most popular antennas for wireless devices is the Inverted-F illustrated in **Figure 5**. The mobile device and Wi-Fi band at 2.4 GHz has a wavelength of approximately 12.5 cm, so a 1/4-wavelength ground-plane antenna would be about 6.25 cm (2.5 inches) long. Even though this is small, it is still too big to be convenient for a phone or other pocket-sized device. As shown in the **figure**, though, the antenna can be bent over and is less than 1 cm above the ground plane at 2.4 GHz. This is a much more manageable proposition for phones and similar items. Freescale Semiconductor's application note "Compact Integrated Antennas"

(cache.nxp.com/files/rf_if/doc/app_note/AN2731.pdf) shows typical inverted-F designs and variations on that theme.

Patch Antennas

Most people think an antenna has to be a wire or tube or rod. Not so! The antenna is only a convenient structure on which current can be made to flow and create the desired radiation pattern. It is the currents that radiate and receive signals, not the conductor on which they are flowing. The patch antenna (and its relative, the slot) is a good example of such an antenna. **Figure 6A** shows a photo of two patch antennas used on the



Your Single Source of Electronic Equipment, Tools, and Components

<p>01DM820B Digital Multimeter Super Economy</p> <p>DMM - Volts, current, Resistance, Transistor and Diode Test. Includes Leads</p> <p style="text-align: right; color: red;">\$7.50</p>	<p>01DMMY64 Digital Multimeter 32 Ranges</p> <p>Including temperature, capacitance, frequency and diode/transistor testing</p> <p style="text-align: right; color: red;">\$29.95</p>	<p>01DMM9803R Bench DMM True RMS</p> <p>Auto and manual range, RS232C standard interface. Runs on DC or AC</p> <p style="text-align: right; color: red;">\$139.95</p>	<p>Sweep Function Generator</p> <p>3MHz, 6 Waveform Functions, Int/Ext Counter, lin/log sweep</p> <p>Model FG-30 (no display) \$135</p> <p>Model FG-32 (5 digit display) \$195</p>
<p>01DS1102E 100MHz Rigol Oscilloscope</p> <p>Features: 1 million points of deep memory, FFT, record and replay, roll mode, alternate trigger mode, and adjustable trigger sensitivity</p> <p style="text-align: right; color: red;">\$399</p>	<p>01SDS1052DL Siglent 50MHz DSO</p> <p>500MSa/s Sampling 32K Memory 6 Digit Frequency Counter USB Interface 7" Color TFT-LCD Screen</p> <p style="text-align: right; color: red;">\$279</p>	<p>DC Power Supplies</p> <p>Model HY3003 Variable Output, 0-30VDC, 0-3Amp. \$95</p> <p>Model HY3003-3 Two 0-30VDC, 0-3Amp Variable Outputs plus 5V 3A fixed. \$195</p>	
<p>Parallax Board of Education Full Kit</p> <p>USB version with Stamp and USB Cable.</p>		<p>BOE Full Kit Parallax 7.5V Power Supply</p> <p style="text-align: right; color: red;">\$98.90 \$8.95</p>	<p>0603ZD98 48W Soldering Station</p> <p>Temp. Controlled 310°F-840°F</p> <p style="text-align: right; color: red;">\$15.25</p>

1 (800) 972-2225

| http://www.elexp.com

| contact@elexp.com

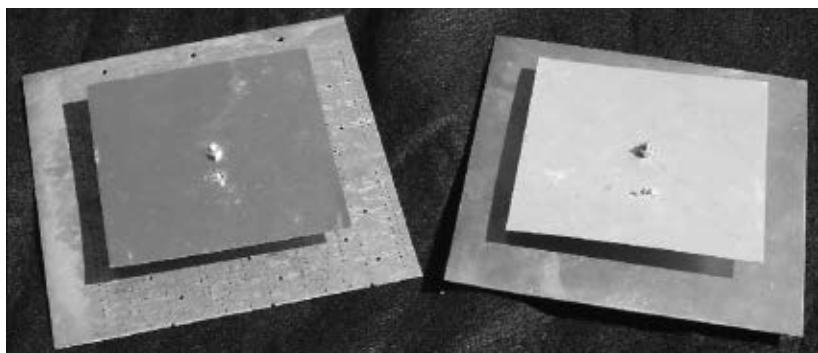
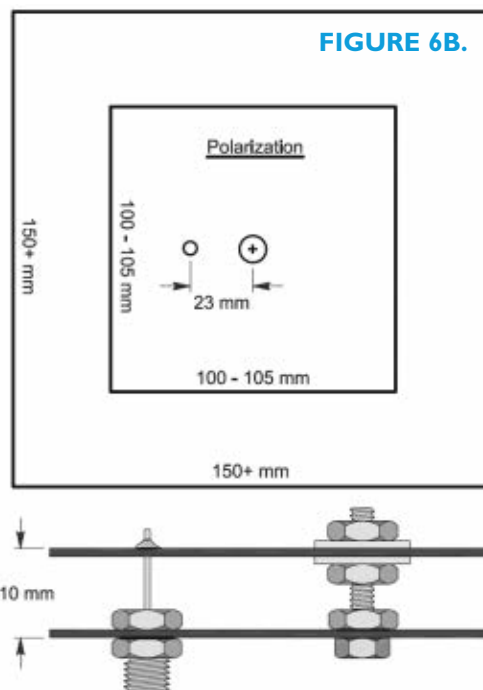


FIGURE 6A. The patch antennas shown here are designed for the amateur 23 cm band and are made from sheet metal (left) and PCB material (right). The basic design information for the antennas is given at B. Polarization is determined by the orientation of the feed point.

amateur 23 cm band near 1,300 MHz. The feed line is attached a small distance from the center of the patch and a bit above an underlying ground plane as shown in **Figure 6B**.

Energy from the feed line excites the entire patch surface, resulting in regions of maximum current. The resulting current patterns create a pattern similar to two parallel dipoles with their currents in-phase and spaced by



approximately $1/2$ wavelength. This creates 2-3 dBi of gain in an omnidirectional pattern above the patch ground plane. Sheet metal or the copper on a printed circuit board can be used so it's easy to see how this type of antenna might be a good choice for a thin flat electronic device!

The Operator as an Antenna

At the start of this column, I suggested that the wireless device user could be an antenna, too, and that's just what happens with a handheld VHF or UHF transceiver such as the popular Family Radio Service (FRS) units. The stubby "rubber duck" antenna provided with the transceiver is a ground plane antenna just like the one described in the first Ham's Wireless Workbench. So, where is the ground plane?

The case of the radio — if metallic — does some of that work, but it's not really big enough to do the job all that well. If the radio's case is plastic, then there is no ground plane at all! Or, is there? Your hand wrapped around the body of the radio makes a very good capacitor, coupling strongly to the radio case or the internal electronics. RF at these high frequencies will flow through that capacitance to your skin and then along the surface of the skin for quite a distance.

Current flowing on a surface radiates and receives whether the surface is metal or a concoction of salt water and protein such as you! While your skin is pretty lossy, that extra antenna surface can make a big difference in communication range.

Can you hear me now? **NV**

Best Value
prototype PCBs online
Turnkey Assembly available!

Serving over 10,000 PCB
Designers and Buyers
since 2003

Made in USA
www.PCBFabExpress.com

☎ 408-522-1500

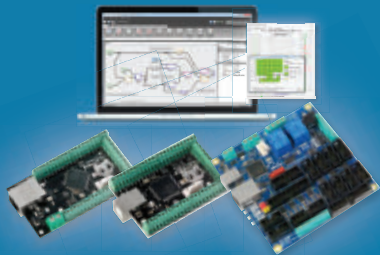
Support3@pcbfabexpress.com

PoLabs

For more information or software download please visit 

www.poscope.com/nv

PoKeys Connect, Control



PoKeys is a successful story about versatile and extremely powerful controller.

Save your time with **fast** connectivity and **easy** configuration of advanced features.

Enjoy rare moments in life while PoKeys is taking care over your house or garden.

Take a break while PoKeys is controlling stepper motors in your CNC or 3D printer.

Let PoKeys make your imagination come true using our intuitive software and plugins for Mach3/4

Stepper motor drivers Drive

Safest and most efficient force for your motors.



PoScope Mega1+ Measure



Do you need a really powerful and portable oscilloscope that will save your time, money and make your life with measurements easier?

PoScopeMega1+ is lowest power consumption USB oscilloscope, function generator and logic analyzer all-in-one available on the market.

MedeaWiz® Sprite

Triggered HD 1080p Video Player

No black when looping!

Serial port 3v or 5v levels

Multiple baud rates



No black when triggered!

Multiple trigger options

Simple serial control

Video / Audio looping with a direct pushbutton / trigger input

Serial port control of up to 200 files - With status feedback

Use for haunted portraits, cue lines, escape rooms, kiosks

sales@TeamKingsley.com

Download manual at www.MedeaWiz.com

phone: (618) 797-9951

Team Kingsley LLC

\$79⁰⁰

We ship world-wide

Mention code "NV" to get this special price until Halloween!

Includes power supply and other accessories



EARN MORE MONEY Get your dream job!

Be an FCC Licensed Wireless Technician!

Make up to \$100,000 a year and more with NO college degree

Get your "FCC Commercial License" with our proven Home-Study Course!

- No costly school. No classes to attend.
- Learn at home. Low cost!
- No experience needed!
- **MONEY BACK GUARANTEE:** You get your FCC License or money refunded!



Your "ticket" to 1,000's of high paying jobs in Radio-TV, Communications, Avionics, Radar, Maritime & more.

Call for FREE info: 800-877-8433 ext. 109

www.LicenseTraining.com

COMMAND PRODUCTIONS • FCC License Training
Industrial Center, 480 Gate Five Rd., PO Box 3000, Sausalito, CA 94966-3000

Our 39th Year

NEW PRODUCTS

■ HARDWARE
■ SOFTWARE
■ GADGETS
■ TOOLS

RAPIDTURN TRANSFORMS CNC MILL INTO LATHE

Tormach, Inc., has announced the release of their RapidTurn™ an innovative 5C CNC chucker lathe attachment designed for use with the company's PCNC 1100 or PCNC 770 mills. Tormach's engineers are excited to bring RapidTurn's affordable and capable turning solution to customers. This add-on accessory allows users to do small turning/lathe operations right on the bed of their mill. With a price tag starting at just \$1,595, this accessory provides the option of getting real turning capabilities without having to purchase a full-size, stand-alone lathe. Product highlights include:

Powered by PathPilot®:

RapidTurn runs on PathPilot: Tormach's approachable yet professional-level CNC machine control system. The RapidTurn utilizes PathPilot's lathe interface which is easy to use, easy to operate, and designed for CNC machinists of all experience levels.

Space-Saving Utility: RapidTurn provides real CNC lathe functionality without taking up precious shop floor space. After initial installation, setup takes just a few minutes. RapidTurn is easily stored out of the way when not in use.

Turning and Milling in One

Setup: RapidTurn has a manual index plate with a locking pin. Parts can be securely positioned in 15° increments for secondary work with the PCNC's primary spindle, making it ideal for cutting wrench flats or drilling cross



holes on turned parts without additional setups.

RapidTurn has a complete suite of accessories available to tackle almost any small turning job, including: quick change toolholders, adjustable tailstock, tooling, three-jaw chuck, and more.

For more information, contact:
Tormach, Inc.
www.tormach.com

EZasPi 40-PIN PROTO BOARDS

Mikronauts is introducing 40-pin GPIO versions of their popular EZasPi series of prototyping boards.

All boards in the EZas line are silkscreened on both sides so that

power and signal busses are highlighted, taking the guesswork out of finding the signals under the solder mask.

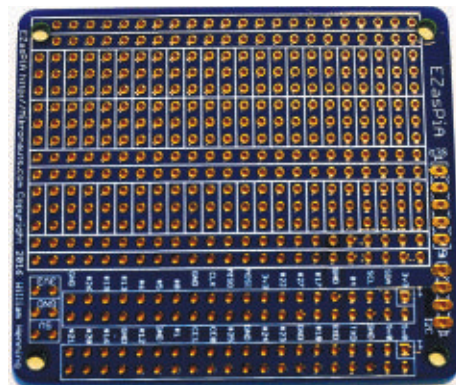
EZasPi boards are designed to minimize the need for jumper wires by careful component placement, and were specifically designed to be easy to prototype with servo headers and screw terminals.

Educational users in particular will find the clear silkscreening to be helpful in teaching environments.

All of the EZas line of prototyping boards are high quality FR4, ROHS, lead free gold immersion (ENIG) boards. Two new products in the EZas line that are designed for the Raspberry Pi are:

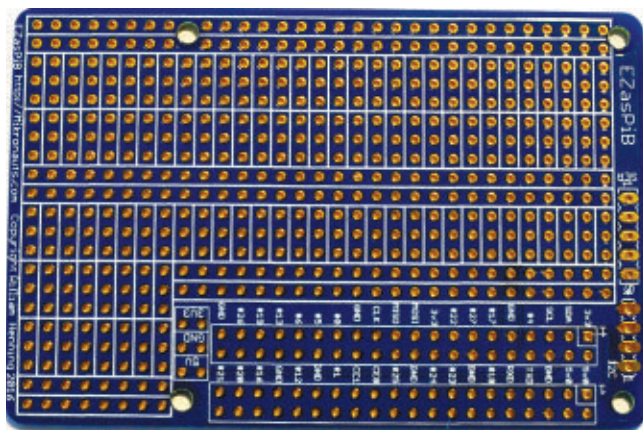
EZasPiA 85 mm x 65 mm

- Pi connector labeled on both sides
- Raspberry Pi A+ mounting holes
- Large prototyping area
- Servo and screw terminal friendly
- EZas silkscreening
- I²C header pads
- COM header pads
- Stacking headers available



EZasPiB 85 mm x 56 mm

- Connector labeled on both sides
- Raspberry Pi B+ mounting holes
- Extra large prototyping area
- Servo and screw terminal friendly
- EZas silkscreening
- I²C header pads
- Stacking headers available



Quantity pricing is available for distributors and educational organizations.

For more information, contact:
Mikronauts
<http://Mikronauts.com>

AC/DC BALANCE (DIS)CHARGER

Hitec is now offering their Power Peak D7 AC/DC Charging Station. Armed with two independent 200 watt output ports producing up



to 20 amps of charge current each, the Power Peak D7 is capable of charging all battery chemistries quickly and efficiently. The D7 also features mini USB sockets, allowing users to monitor battery health using the latest Logview Software, and stay current with convenient software updates. The 5V/1A USB charge socket provides compatibility with smartphones, digital cameras, and other USB-powered devices.

Multiplex's integrated BID system brings convenience and safety to the Power Peak D7, with its worry-free programming for batteries with the optional BID keys and chips. Essential pack configuration data is

automatically passed to the charger, allowing users to simply plug in a battery and start charging without any further setup effort. The Power Peak D7 has two separate backlit graphic screens for readability and four language choices: English, German, French, or Italian. It is priced at \$274.99. Some of the features and specifications include:

- Input Power Source: 110-240V AC/11-30V DC
- Total Maximum Output: 400W
- AC Input: 110-240V AC
- DC Input: 11-18V DC
- Charge Circuit Power: 200W x2
- Charge Current Range: 0.1-20.0A x2
- Discharge Current Power: 36W x2
- Discharge Current Range: 0.1-10.0A x2
- Current Drain for LiPo Balancing: 300 mA per cell
- NiCd/NiMH Battery Cell Count: 1-18 cells
- LiPo/LiFe/Lilon Cell Count: 1-7 cells
- Pb Battery Voltage: 2-24V
- Net Weight: 92 oz (2.6 Kg)
- Dimensions: 10.6 x 7.9 x 3.3 in (270 x 200 x 85 mm)

For more information, contact:
Hitec RCD USA, Inc.
www.hitecrd.com



GUI MADE EASY

The arLCD 3.5" smart TFT touchscreen combines Arduino UNO with the ezLCD smart LCD.

earthlcd.com/nuts

SDP Stock Drive Products
Setting Ideas Into Motion

One-Stop Shop for
Mechatronic Components

EXPLORE
DESIGN
BUY ONLINE



www.sdp-si.com
no minimum requirement

Designatronics inc.

ROBOT POWER Extreme Motor Speed Control!

MegaMoto GT- Mega Motor Control for Arduino™

- 6V-40V - 35A / 50A+ peak
- Single H-bridge or dual half
- Current and Temp protected
- Jumper select Enable & PWM
- Robust power terminals



The VYPER™

- 8V-42V - 125A / 200A peak!!
- Single H-bridge
- RC and Pot control
- Overload protected
- For your BIG bots

www.robotpower.com

Phone: 360-515-0691 • sales@robotpower.com



New in the
Nuts & Volts
webstore!

If you've ever looked at an electronic gadget and thought, "How does that work?" or "How can I make that?" Then look no more.

Whether you're 8 or 100 years old, as long as you're curious and have a playful spirit, this 328 page book is for you.

For complete details and to view a sample chapter from the book, visit our webstore @ store.nutsvolts.com

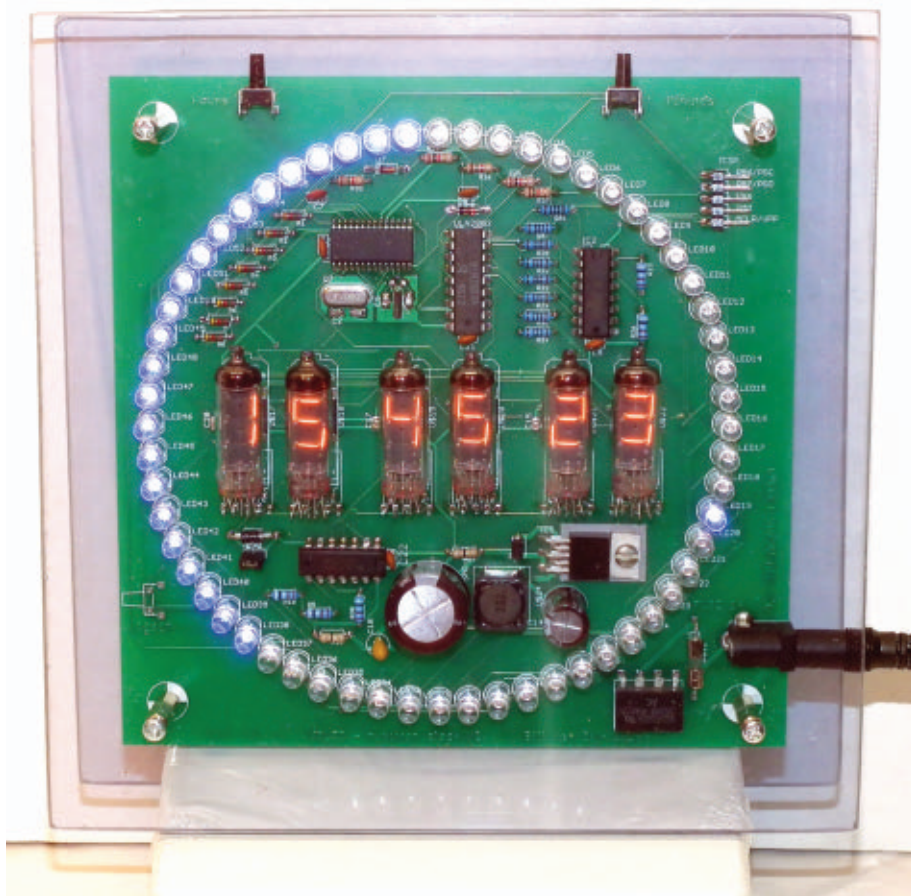
Only
\$24.95

Build the Numitron

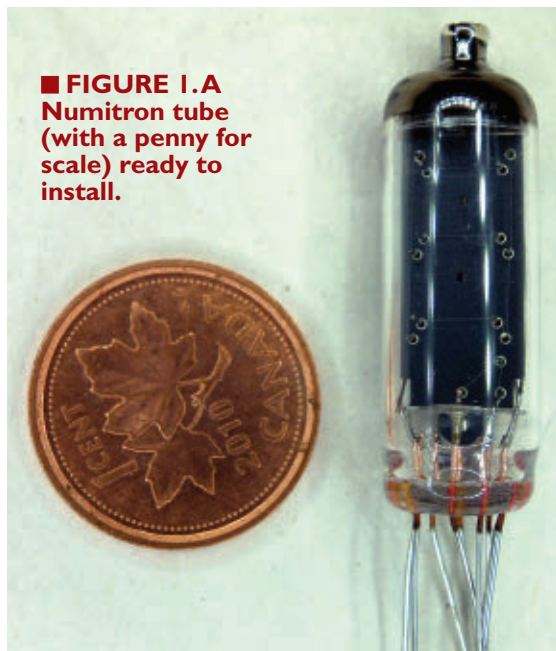
A SIX-DIGIT CLOCK

By Bill van Dijk

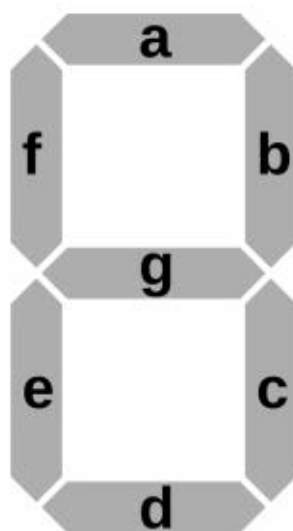
Post comments on this article and find any associated files and/or downloads at www.nutsvolts.com/magazine/article/September2016_Numitron-LED-Clock.



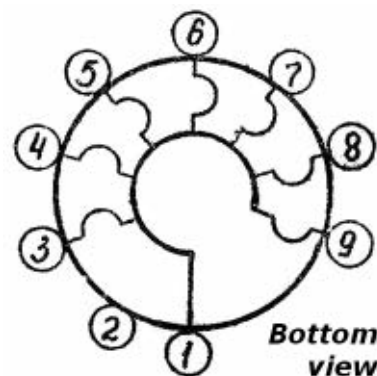
I suspect that for many of you the name Numitron is new, while for many others it will bring back (hopefully fond) memories. The venerable electron tube (a.k.a., thermionic valve) has all but disappeared from the mainstream electronics world, and along with them related old technologies such as the Nixie tube (featured in previous *N&V* articles), Dekatrons, Numitrons, and Panaplex displays, and (to a lesser extent) Vacuum Fluorescent Displays (VFDs) have also faded into the background.



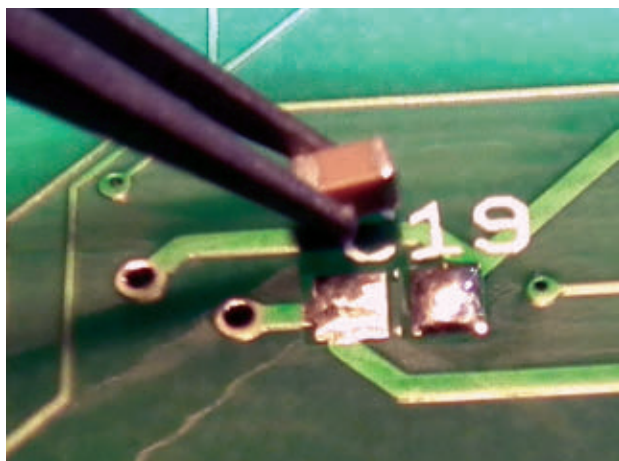
■ **FIGURE 1A**
Numitron tube
(with a penny for
scale) ready to
install.



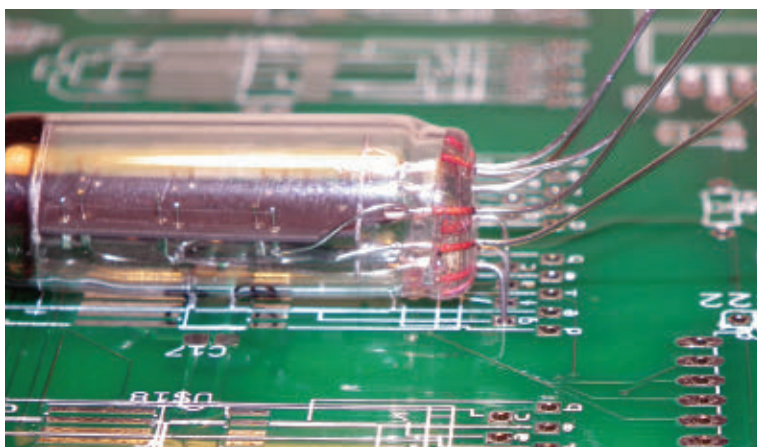
■ **FIGURE 2A.** A typical seven-segment display layout uses letters to identify each individual lighted part of the display.



■ **FIGURE 2B.** The pinout of a Numitron tube. Note that since the Numitron uses filaments, the common can be positive, negative, or even power via AC voltage.



■ **FIGURE 4.** Close-up of placing one of the 0805 caps on the board.



■ **FIGURE 3.** The Numitron tube mounts directly to the PCB without the need for a socket.

This particular project was designed to showcase Numitron tubes (**Figure 1**), which are still available for a reasonable cost on eBay from old Russian cold war stock. As stocks (rapidly) dwindle, their cost increases steadily until they will be gone forever. Don't miss out on this opportunity to build a useful project showcasing this beautiful cold war era component.

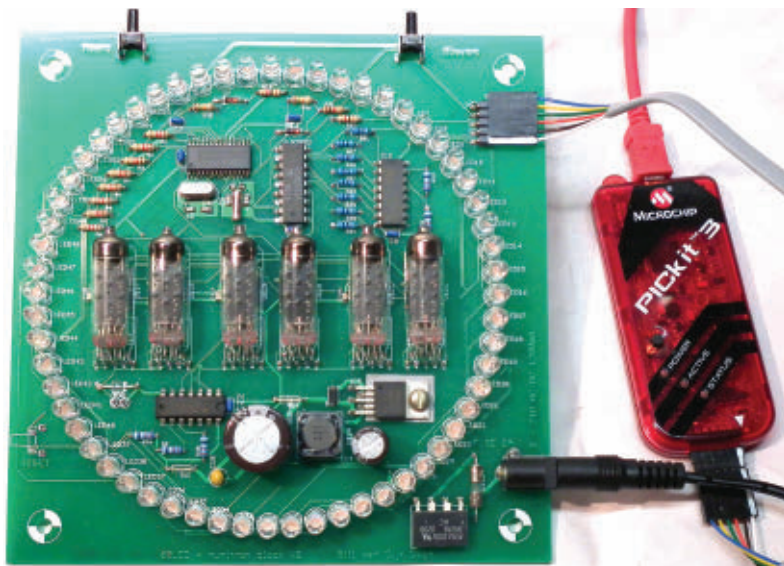
The Numitron is not a traditional tube (valve) in that there is no electron emission involved; it only has filaments, and as such is more analogous to the light bulb than any other electrical device. Seven small filaments (**Figure 2a**) are arranged in the traditional formation of the seven-segment LED numerical display we are all familiar with (**Figure 2b**), all within a simple glass tube. The particular Numitron used in this project (the IV-9) operates on five volts at about 23 mA per segment. It has "fly

leads" and as such does not require a difficult socket (**Figure 3**).

The Project

In order to effectively showcase the Numitron, I wanted to do it in a useful as well as attractive way. I also wanted to integrate it with more modern technologies — a hybrid as it were — and use as many parts possible that I had in stock. I love challenging my soldering iron, so SMDs (surface-mount devices) were used as well (**Figure 4**).

The "brains" of the project is a Microchip PIC, and it is a very busy fellow in this project; it's in control of everything happening on the board. LEDs lend themselves very well to multiplexing, and that is used extensively to control the 60 LEDs that form the circle around the



■ **FIGURE 5.**The PICkit3 connected to the Numitron board for programming.

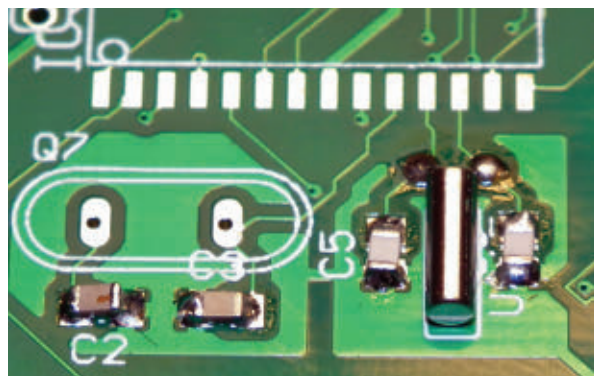
Numitrons. They display eight different programmed patterns, most based on a one minute repeat cycle.

The Numitrons, on the other hand, do not lend themselves at all to multiplexing, and therefore a BCD decoder (with latch) was used to drive each tube. I'll explain more about the operation of those later on. The schematic and board were created in Eagle, and the files are available for download at the article link in native Eagle format as well as the Gerber RS274x format.

The PIC

Programming of the PIC is done through the ICSP connector located on the board. I use the PICkit[™]3 combined with MPLAB IPE (integrated programming environment); refer to **Figure 5**. Other options are available for programming, although programming SMD devices off-board often requires a special adapter. The main system crystal frequency is 4 MHz. The clock (time) crystal is a small 32 kHz watch crystal. *NOTE: This is a critical part for clock accuracy!* Please resist using an unknown or salvaged part here.

The two 12 pF load capacitors (**Figure 6**) determine



■ **FIGURE 6.**The clock timing crystal installed on the PCB with the surface-mount noise decoupling capacitors.

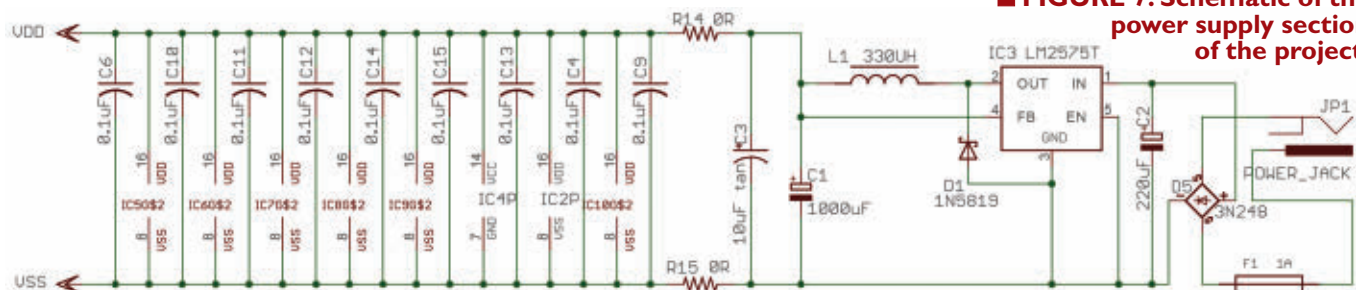
its frequency accuracy, and must be matched with the part. Accuracy of one second per day or less is possible if the crystal and load capacitors are properly matched. If you use a different crystal than listed, be sure to change the load capacitor values to the specification of your part.

A 0.1 μ F ceramic capacitor is placed close to the PIC (as well as the other chips) to provide noise decoupling. After all, there is a lot of heavy duty switching happening on this board. The reset button is not debounced (not needed), and for all intents and purposes may be left off the board all together. Simply disconnecting and reconnecting the power will do the same, and prevent non-intentional reset when in use.

The Power Supply

The power supply is fairly simple, using a switch mode buck converter to provide the five volts required for the clock (**Figure 7**).

Though I considered a linear regulator, the part became hotter than I was comfortable with while testing in a circuit like this. Additionally, a switch mode supply allows for more choices when it comes to wall wart style power supplies (**Figure 8**). The greatest current requirement is during the start-up flash cycle, where all six Numitrons are lit for a second at the time; 7x6 segments at about 23 mA = 966 mA total. Maximum current



■ **FIGURE 7.**Schematic of the power supply section of the project.

required for the project peaks therefore very close to the one amp limit when all Numitron segments are lit when first plugged in (or after reset).

The bridge rectifier allows for an AC wall wart to be used, or to allow for different DC wall wart polarity options. You may decide to leave the bridge rectifier out and jump the appropriate points on the board to match your wall wart polarity. Of course, you may also use a different power jack than specified; these are perfect parts to be salvaged from old discarded equipment.

Be sure the wall wart can supply at least 1.5 amps of current at minimum nine volts. The 1,000 μ F tank capacitor is a low profile unit so it does not interfere with the front panel. If you use a taller part, you may have to adjust the front panel spacers. The inductor is also an SMD part selected for space reasons. There's a special mounting tip discussed in a more detailed construction manual that is available at the article link.

The LEDs

The LEDs are multiplexed in seven rows of eight, with one additional short row of four for a total of 60. The PIC drives one row at a time, and at a couple of mA each, the

current draw is well within the specifications for the PIC. By using one byte per row, there are never more than eight LEDs lit at any given time (**Figure 9**). The circle will appear to be fully lit when required, thanks to a fast multiplex rate and a phenomenon called 'persistence of vision' in our eyes.

It is important to use high output (also referred to as ultra-bright) LEDs for this design since regular LEDs will appear too dim. You may choose any color you like; the slight difference in voltage requirement for the different colors is insignificant in this project, and does not require any further consideration. Since most of these high output LEDs are "water clear" units, placement is very important to provide a visually pleasing result (**Figure 10**). The construction manual at the article link provides tips for proper installation and alignment.

While a PIC pin will only

need to drive a single LED at any given time, the combined current of the full row may need to be sunk in order to light all eight in that row since this exceeds the current output capacity of the CD4017. The ULN2803 acts as the driver there. Also, we need a way to select the correct row synchronized with the data sent from the PIC port. This is the task of the CD4017 decade counter.

It counts the pulses sent from the PIC and activates the appropriate Darlington pair in the ULN2803. After eight pulses, the CD4017 is reset by the PIC, and is ready to start from the beginning. This arrangement allows any individual LED to be addressed by the PIC, respecting all maximum current limits.

The Numitrons

The Numitrons are essentially just little multi-filament

Resources

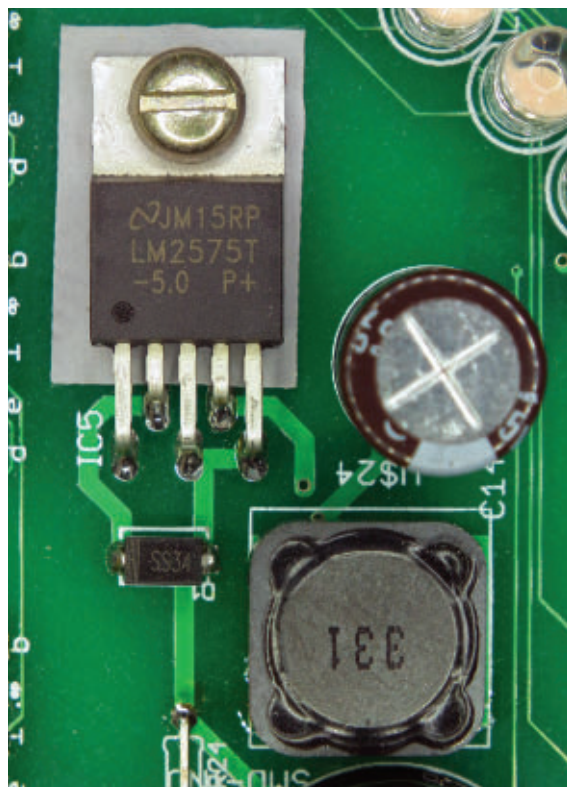
Debouncing Discussion
www.eng.utah.edu/~cs5780/debouncing.pdf

Free Version of EagleCAD
<https://cadsoft.io>

PICKit3 IDE
www.microchip.com/pagehandler/en-us/family/mplabx/home.html

Numitron Lifespan
www.rfcafe.com/references/popular-electronics/images2/rca-dr2010-numitron.pdf

Cathode Poisoning
www.tube-tester.com/sites/nixie/different/cathode%20poisoning/cathode-poisoning.htm



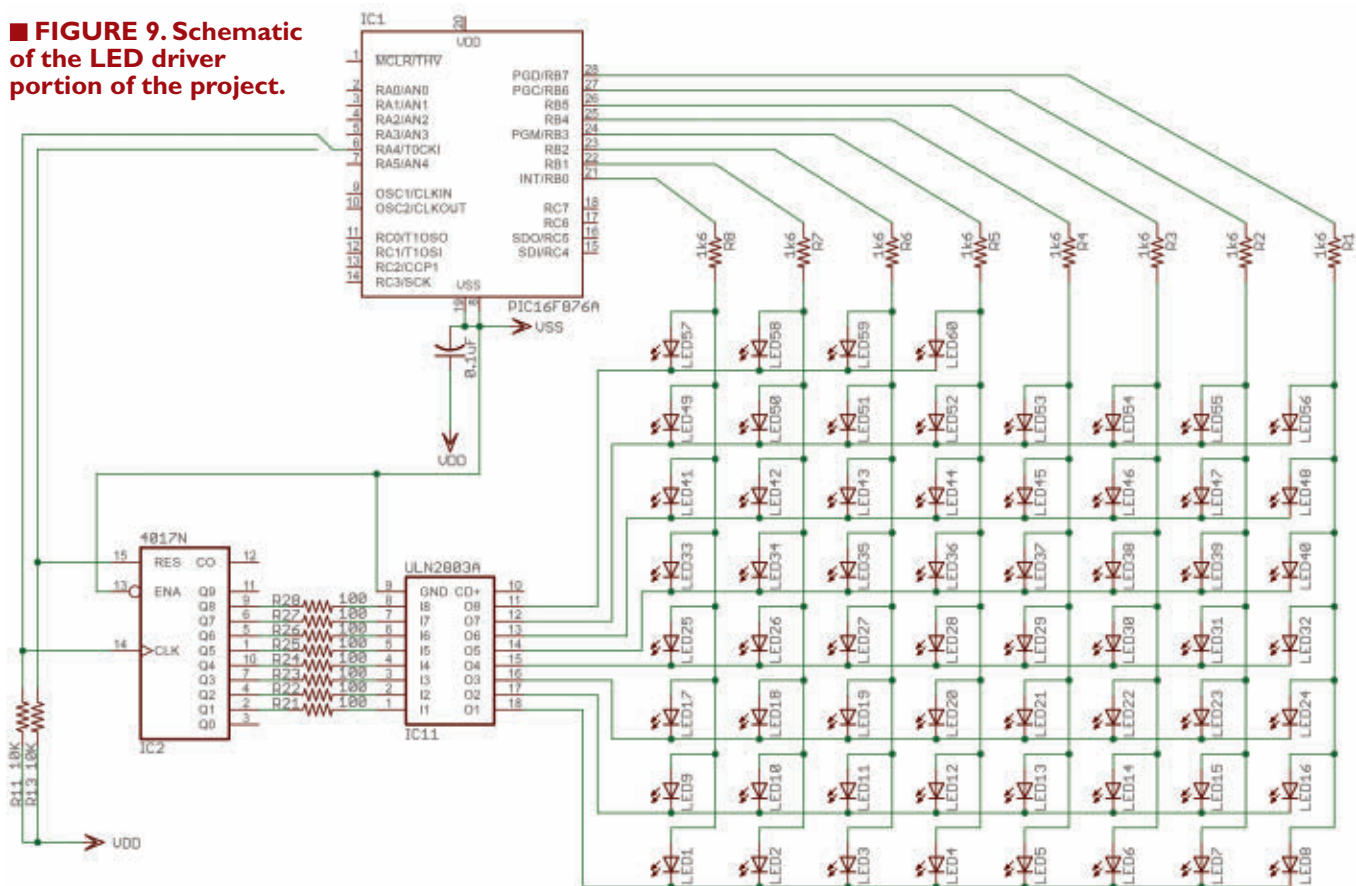
■ **FIGURE 8. Close-up of the LM2575T regulator installed on the PCB.**

The manufacture of the first Numitron was — according to Wikipedia (https://en.wikipedia.org/wiki/Seven-segment_display) as well as a March 1970 *Popular Electronics* article (www.rfcafe.com/references/popular-electronics/numitron-readout-march-1970-popular-electronics.htm) — attributed to RCA. It was built based on the common seven-segment display format first patented in 1908 by someone called F.W. Wood. Around the same time the Numitron was created, LEDs were also up and rising, and quickly posed a significant competitor.

The Numitron had an early advantage: At that time, they were slightly brighter than LEDs of those days, and did not have a great dependence on ambient temperature. Nixies, Panaplex displays, and later LCDs required heating elements when used outdoors in colder regions. Even today, the Numitron is still regularly found working in older gasoline pumps.

The Numitron is still a vacuum tube, and the wires of the IV-9 pass via a sealed hole through the glass envelope. Great care must be taken to ensure no stress (from bending or soldering) is transferred to the glass, which may cause it to crack.

■ **FIGURE 9. Schematic of the LED driver portion of the project.**

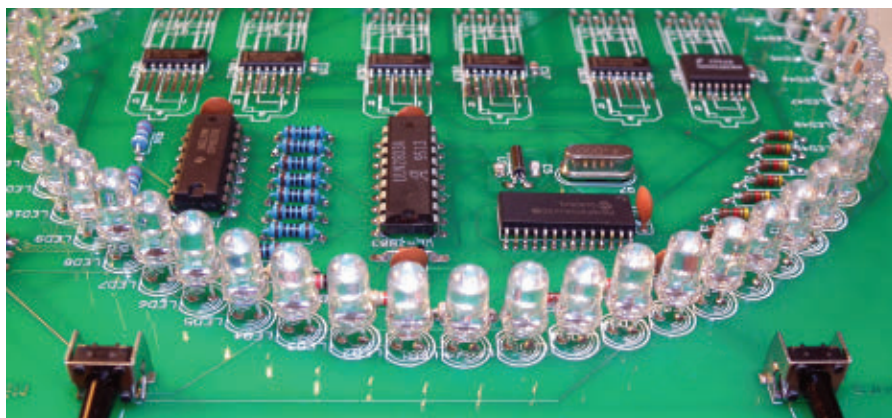


incandescent light bulbs from an electrical perspective, and as such, multiplexing them will simply act as a light dimmer. A 1x6 (or even a 2x3) multiplexing arrangement would leave them very dim, so is not a viable option. Luckily — since their current requirement at about 23 mA per segment is fairly low and it is a seven-segment device operating at about five volts — we can drive them with a standard BCD to seven-segment decoder designed for seven-segment LED displays with a max driving output of

25 mA. The CD4511 also has a data latch built in, so it was chosen for this design.

I opted for the SMD version since they neatly tucked behind the Numitron tube, making for a nice clean appearance (**Figures 11a** and **11b**).

Since I am now rapidly running out of ports on the PIC, I needed an efficient way to control the CD4511s and their latches. This is done by the 74164 shift register in a similar fashion as to how the CD4017 operates in the

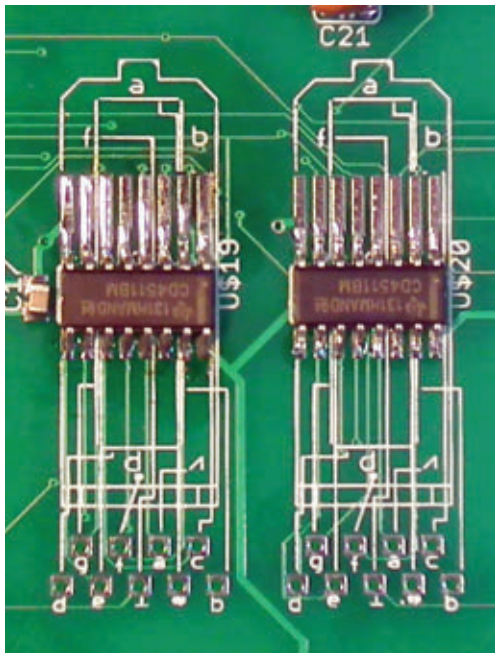


■ **FIGURE 10. Close-up of the top of the PCB showing the water-clear LEDs in place.**

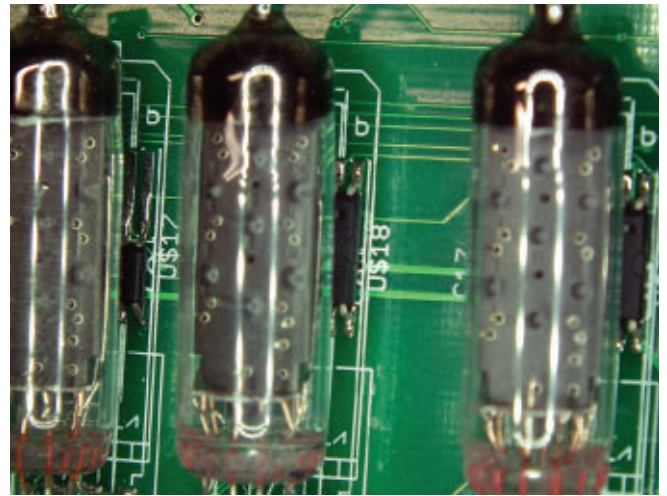
LED section. The BCD data is presented to all the CD4511s at the same time via the data bus, but decoded and latched only by the CD4511 selected by the SN74HC164 shift register.

Synchronization of data and latch selection is controlled in software.

Since the time only changes every second, only one Numitron digit is updated for each time the LED circle is updated. This made the software easier, and it is still much faster than actually required. The CD4511 also has a “lamp test” function which is used to flash all segments during startup and for reset of the clock, also providing an easy way to



■ **FIGURE 11A.**
Close-up of the
surface-mount
CD4511 BCD to
seven-segment
display decoder.



■ **FIGURE 11B.**
Numitron tube
mounted
snugly atop the
surface-mount
CD4511 chip.

test the Numitrons.

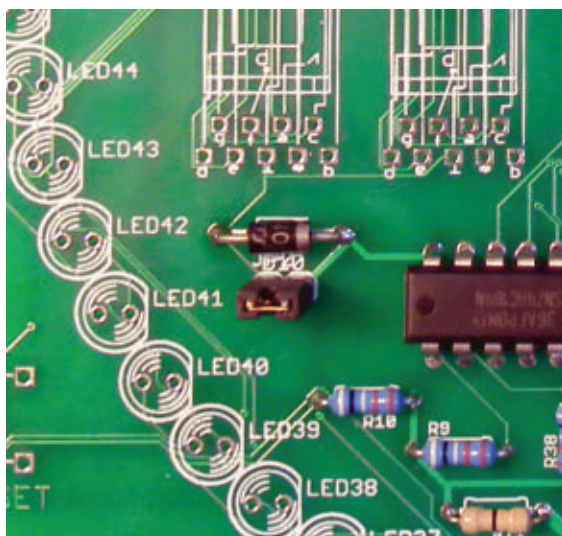
Note also the diode D2 and jumper JP2 (**Figure 12a**). They were intended to be a rudimentary option for lowering the brightness of the Numitrons and lower the current requirement slightly. It simply relied on the forward voltage drop of the diode, with a jumper to short it (**Figure 12b**). It is not really required and may be left off the board, in which case a simple jumper wire may be inserted in place of the diode.

Buttons

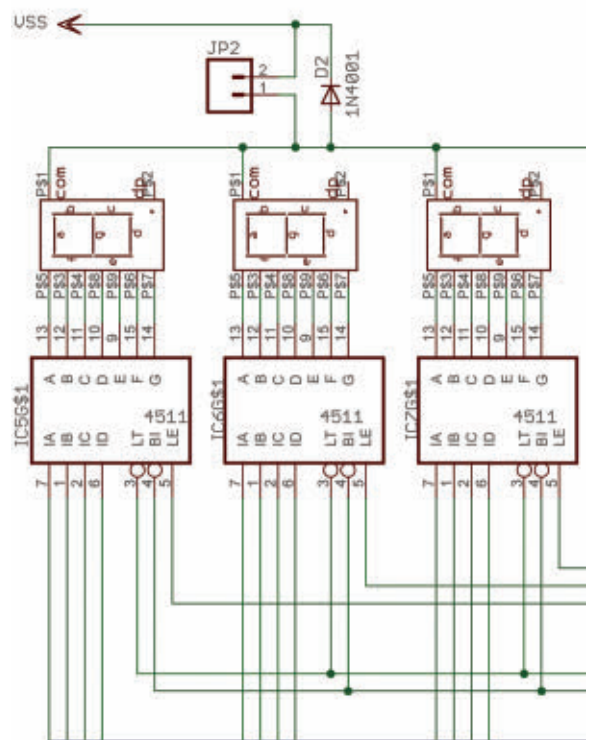
The clock is set using only two buttons with multiple functions to control 12/24 hour display selection, LED pattern selection, as well as setting the time.

Buttons (or switches) are difficult things to read in high speed digital circuits since the mechanical parts in a switch or button actually bounce like a dropped ball; it takes a bit of time

■ **FIGURE 12A.**
Location of the
diode and jumper
JP2 on the PCB.

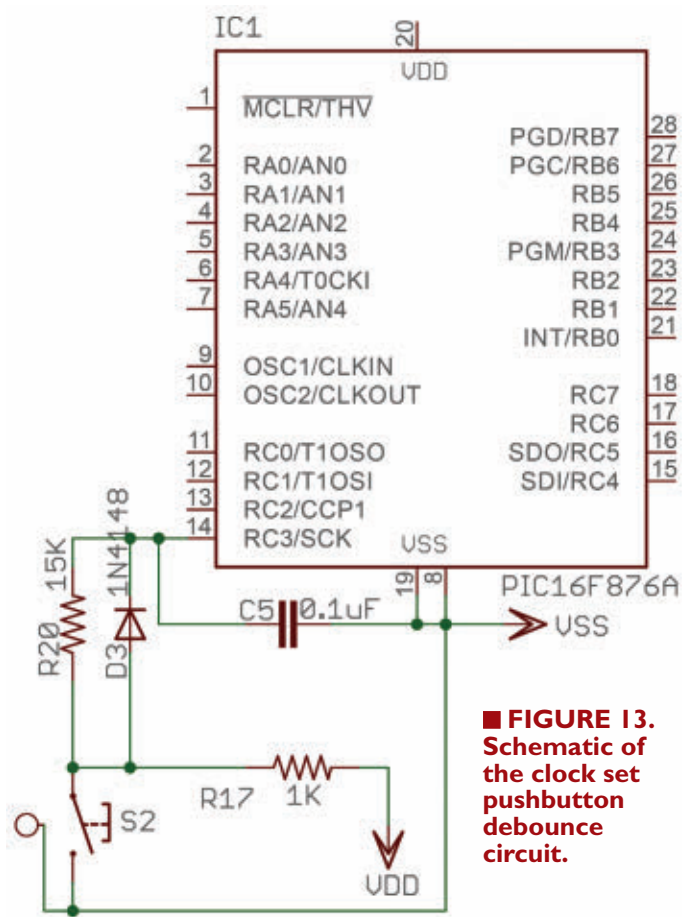


■ **FIGURE 12B.**
Schematic
showing the
location of the
jumper and
voltage dropping
diode.



to stop settling in one state or the other when pushed. Debouncing can be done in software, but in this case, a hardware solution was used. The solution is based on the RC time required to charge/discharge a small capacitor (**Figure 13**).

When the button closes, the capacitor (C7) is discharged through a resistor (R19), allowing the bounce to settle before the voltage on the PIC input pin falls below the logic level 0. When released, the capacitor is charged quickly through R18 and D4. By using this method, no further precautions are required in the software, and the button is treated like an ideal button without bounce.



■ **FIGURE 13.** Schematic of the clock set pushbutton debounce circuit.

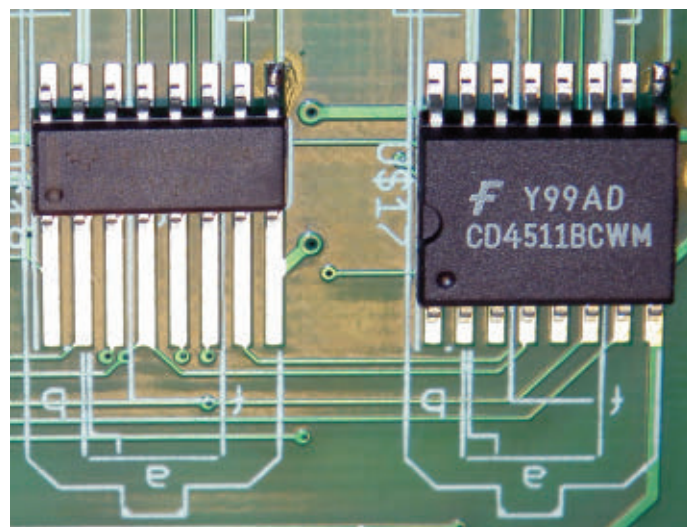
A fantastic discussion on the subject is listed in **Resources**.

The Printed Circuit Board

The printed circuit board (PCB) is a standard two-layer version, and has no complicated manufacturing issues for most board houses. As mentioned, the Eagle files as well as Gerber (RS274x) files for those board houses that do not accept native Eagle files are at the article link. Of interest is the somewhat odd looking footprint for the CD4511s (**Figure 14**).

These chips come in wide and narrow packages, and in the past I have received either or both mixed in my orders. Functionally either one is acceptable, so I created an Eagle footprint that would accept both wide and narrow formats. This is shown in greater detail in the download package for this article.

The circle of LEDs was created by specifying a set of vector coordinates in Eagle by specifying the center, radius, and angle for each LED. The board is about 6" square, and has four mounting holes in which stand-offs are used to hold the front and rear Plexiglas or acrylic



■ **FIGURE 14.** Example of a PCB that can accommodate both wide and narrow versions of the SMD chip.

panels which allows the finished product to stand vertical.

A free version of Eagle CAD can be downloaded from the link in **Resources**.

Software

The software is written in MPASM (absolute code) — an assembly language for the PIC. The development IDE (integrated development environment) includes the programming software for the PICkit3 mentioned earlier, and can be downloaded for free from the Microchip website (see **Resources**).

The source code is available at the article link. The code is heavily commented, so it will help those who wish to dig further into that aspect of the project. Aside from the rather simple section of code that runs the clock, there is a LOT of stuff involved in running the LED displays. This PIC has two eight-bit internal timers and one 16-bit timer, which each generate an interrupt when overflowing. One is used for the time, overflowing every second to advance the clock; the second is used to schedule the display updates; and the third is used for the LEDs.

In the various LED patterns, a "dot" flies around the circle, filling each circle one second at a time. At the 60 second mark, all dots are lit and reset at the new minute. The dot at the start of the minute must make 60 steps to reach the end, and each subsequent dot makes one step less than the previous one. That means the timing of each step changes with each subsequent second.

Timer 2 has a special register (PR2) that allows the timer period to be changed on-the-fly; the timing values required for each step are stored in lookup tables. The positions of the dot are also stored in a lookup table. Setting preferences (12/24 mode and preferred LED

pattern) are stored in the PIC's internal EEPROM, and are therefore not lost on reset or power failure.

General Notes

Building this project is probably at an advanced skill level, mostly due to the number of surface-mount parts. Even if you don't intend to build this clock, you may enjoy reading the

Soldering SMD Components

Small components are best soldered with small tools. That means a very fine soldering tip, fine point tweezers, and very thin solder. The SMD components in this project are not very difficult to solder, however, I'll describe the methods I use.

Caps, resistors, etc., come in many sizes; 0805 (used in this project) and larger can easily be soldered as follows: Start with placing a small drop of solder on ONE pad. With the tweezers, place the part on the pad, heat the drop, and push the part in place (refer to **Figure 10b** in the text). If the part is not located properly, reheat the one side and correct. Once properly placed, solder the other side.

A similar method is used for the ICs. Place solder on ONE pad, and place the part. If not located correctly, reheat and reposition. Do NOT push to bend the leg even a little bit. The legs will break easily, and the stress will add to future failure chances (**Figure 13a**). Please see the construction manual at the article link for more in-depth information.

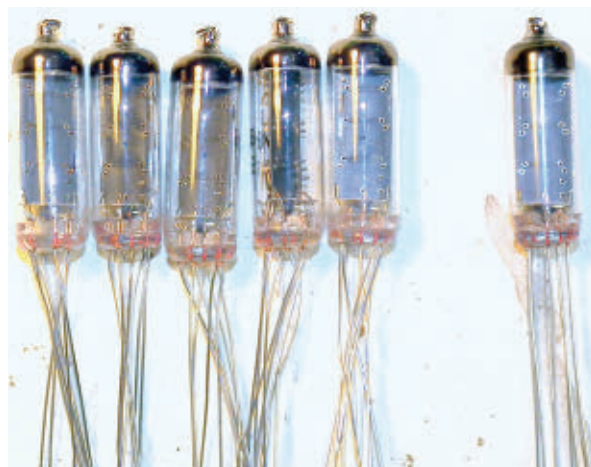
PART	QTY	VALUE	DEVICE	VENDOR
C1	1	1000 µF/35V	Capacitor, polarized, 16 x 15 mm	Mouser (647-URS1V102MHD)
C2	1	220 µF/50V	Capacitor, polarized, 10 x 16 mm	Mouser (667-EEU-FM1H221)
C3	1	10 µF tan/16V	Capacitor, polarized, tantalum	Mouser (581-TAP106K016SCS)
C4-C9	6	0.1 µF	Capacitor, ceramic, through hole	Mouser (75-1C10Z5U104M050B)
C10-C15	6	0.1 µF	Capacitor, ceramic, 0805 SMD	Mouser (581-08055C104K)
C16 + C17	2	12 pF	Capacitor, ceramic, 0805 SMD	Mouser (581-08055A120J) *
C18 + C19	2	20 pF	Capacitor, ceramic, 0805 SMD	Mouser (80-C0805C200J5G)
D1	1	1N5819	Schottky diode, high speed	Mouser (512-1N5819)
D2	1	1N4001	Diode, rectifier	Mouser (512-1N4001) *
D3 + 4	2	1N4148	Diode, switching	Mouser (512-1N4148)
D5	1	3N248	Diode bridge 1.5A	Mouser (625-3N248-E4) *
F1	1	1A	Fuse, 10x3.8 mm	Mouser (576-07771.25MRET1P) *
IC1	1	PIC16F876A-I/SO	Eight-bit Microchip PIC (SO28W)	Mouser (579-PIC16F876A-I/SO) *
IC2	1	CD4017BE	Decade counter (DIL16)	Mouser (595-CD4017BE)
IC3	1	LM2575T-5	Switch mode regulator 5V (TO220-52)	Mouser (863-LM2575T-5G)
IC4	1	SN74HC164N	Eight-bit shift register (DIL14)	Mouser (595-SN74HC164N)
IC5-10	6	CD4511BPW	BCD to seven-segment driver/latch (SMD)	Mouser (595-CD4511BPW) *
IC11	1	ULN2803A	Darlington transistor array (DIL18)	Mouser (511-ULN2803A)
JP1	1	POWER_JACK	Power jack, 7x2.1 mm	Your favorite/various *
JP2	1	Two-pin	Straight 0.1" spacing with jumper block	Your favorite/various *
JP3	1	Five-pin	Angled 0.1" spacing	Your favorite/various
L1	1	330UH	SMD power inductor	Mouser (851-CDRH127NP-331MC)
LED1-60	60	UltraBrite LED	LED 5 mm	Your favorite/various *
NU1-6	6	IV-9	IV-9-Numitron	eBay *
R1-R8	8	1k6	Carbon resistor, 1/4W, axial lead, 5%	Your favorite/various
R9-R13	5	10K	Carbon resistor, 1/4W, axial lead, 5%	Your favorite/various
R14 + R15	2	0R	Use jumper wire	Your favorite/various
R16-R18	3	1K	Carbon resistor, 1/4W, axial lead, 5%	Your favorite/various
R19 + R20	2	15K	Carbon resistor, 1/4W, axial lead, 5%	Your favorite/various
R21-R28	8	100 ohm	Carbon resistor, 1/4W, axial lead, 5%	Your favorite/various
S1-S3	3	Switch	Long tactile button, 90 degree	Mouser (506-FSMRA4JH04)
X1	1	4 MHz	Crystal, low profile	Mouser (520-040-20-4X-DU)
X2	1	32 kHz	Watch crystal	Mouser (732-C002RX32.76K-APB) *
Circuit board	1	N/A		Favorite boardhouse
Wall wart	1	12V or 15V, 1.5A, AC or DC	(Connector to fit JP1)	Your favorite/various
Spacer	4	M3x20 mm F-F		Your favorite/various
Spacer	4	M3x15 mm M-F		Your favorite/various
Screw	8	M3x10 mm		Your favorite/various
Panel	2	8" x 8" x 1/4" acrylic		Local plastics shop *

Parts denoted with * are specifically mentioned in the text. Please read before ordering.

**PARTS
LIST**



■ **FIGURE 15.** The tools I use for surface-mount components on PCBs.



■ **FIGURE 16.** Typical set of Numitron tubes available on eBay.

construction manual since it contains many tips and examples that may be of use to you in this or other projects. None of the SMDs used here are ridiculously difficult to solder by hand, but it does require a steady hand, good light, a magnifying glass, and a good soldering iron.

My Weller soldering station has replaceable tips, and the tip I use for the SMD parts is almost needle sharp. Combined with very thin solder (0.020"), it is really not too hard to do (**Figure 15**). I have included additional tips in the construction manual at the article link. When ordering the SMD capacitors, order a few spares. If you lose an SMD capacitor during construction, look at the end of your soldering iron first. If it's not there, best go get another one.

I usually buy things like the small SMD parts in kits from online providers; it provides lots of spares that way. Speaking of online providers, eBay is a good source for inexpensive parts if you are not in a rush. Since you will probably get your Numitrons there (**Figure 16**), other parts can be ordered at the same time. LEDs can be found on eBay for very cheap, but it's definitely "buyer beware" in my experience; many are junk.

Don't buy the "1,000 for a dollar" deal because you **will** be disappointed. I have included Mouser part numbers for many of the components if you prefer not to take chances. Of course, there are many other excellent vendors to choose from as well (check the advertisers in this issue!).

I am considering helping out those who need assistance with this project. This could be with programmed PICs, or perhaps boards that are partially assembled (SMD parts) on an individual "first come, last served" basis. (All I can promise for sure is that although I'm not very good, I am definitely slow!) You can contact me at TheOldPhart2@gmail.com if you have any questions or suggestions.

Another important thing to keep in mind is static protection — especially in the winter when houses become dry, and static discharges (even so small that you can't feel them) are guaranteed to destroy or damage many of the parts in this project.

Numitron vs. Nixie?

So, now that you've had a chance to look over this clock, you might be considering building one ... but maybe you had your heart set on a Nixie tube clock. Though Nixies are very cool and retro in their own right, the Numitron tube clock has the same feel while offering some distinct advantages over Nixies:

- They are low voltage devices, so — unlike Nixie tubes — you do not need to insure carefully insulated enclosures to keep users from getting a nasty high voltage "bite."
- They do not suffer from failure states such as "sputtering" where electrode metal collects on the inside of the glass tube dimming or obscuring the display.
- They don't encounter cathode poisoning that requires a potentially distracting anti-cathode poisoning software routine.
- In many cases, Numitrons will outlast Nixie tubes as they are typically rated for over 100,000 hours (~11.5 years) of operation compared to some Nixies that are rated for only 5,000 hours (~200 days!).

All in all, Numitrons are a fun and interesting item, and having them used in a clock while surrounded by high tech LEDs makes for a great conversation piece! I highly recommend Numitrons even if you already have a nifty Nixie clock.

If you do end up building a Numitron clock of your own, please let us know! **NV**

Vintage Computing

Simple Hardware Interfaces for Your Mentor's Friend

By Dane Weston
daneweston15@gmail.com

Post comments on this article and find any associated files and/or downloads at
www.nutsvolts.com/magazine/article/September2016_RetroComputer-Simple-Hardware-Interfaces.

Previous articles in Nuts & Volts introduced the Mentor's Friend: a Propeller-based retro computer that you and a young protégé can build together and program in BASIC. Most of the projects in these initial articles focused on the fundamentals (with emphasis on "fun") of Color BASIC which is the software "operating system" inside the Amigo retro computer. This time, we'll venture outside Color BASIC to interface with a couple of simple hardware circuits. So round up your young students, fire up your Amigo, and let's get started!

We'll begin the adventure with the Experimenter's Section of your Amigo which has a small breadboard, power and Propeller I/O pin headers, and a few switches and LEDs — everything you need to get started except a few jumper wires and basic electronic components. If you don't have a well-stocked goody box or a favorite component supplier, one convenient source for parts is the Parallax web store. The Parallax "Boe-Bot and Shield-Bot Refresher Pack" (product ID 572-28132) contains a couple of tact switches, LEDs, resistors, capacitors, jumpers, and several other goodies you can use with your Amigo.

We'll use the program BLINK.BAS from the December 2015 article as our starting point. If you previously saved this code, LOAD it from your SD card and then RUN it. If you don't have the program saved, type NEW at the flashing prompt, then press <F1> for the editor and enter this code:

```
10 CLS
20 PRINT "AMIGO SIMPLE I/O": PRINT " "
30 PRINT "CONNECT S1 TO P8"
40 PRINT "CONNECT D1 TO P9"
50 PRINT "PRESS S1, AND D1 SHOULD BLINK"
60 REM ~~~ LOOP TO CHECK FOR BUTTON PUSH ~~~
70 IF INA[8]=1 THEN GOSUB 100 REM - '1'=PUSHED
80 GOTO 70
90 REM ~~~ SUBROUTINE TO BLINK LED ONCE ~~~
100 OUTA[9]=1 REM - '1'= LED ON
110 PAUSE 250 REM - WAIT 250MS
130 OUTA[9]=0 REM - '0'= LED OFF
140 PAUSE 250 REM - WAIT 250MS
150 RETURN
```

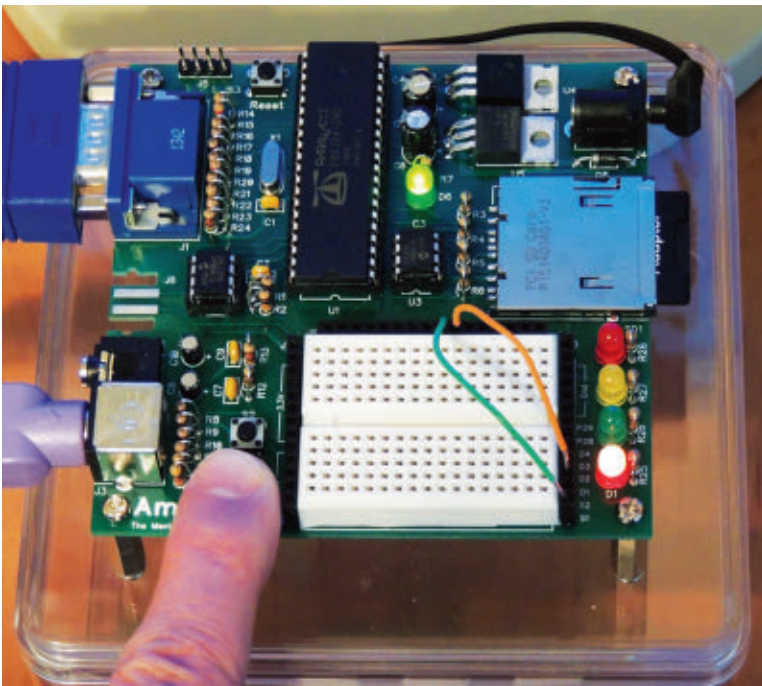


FIGURE 1: A blinking LED using the Experimenter's Section of the Amigo.

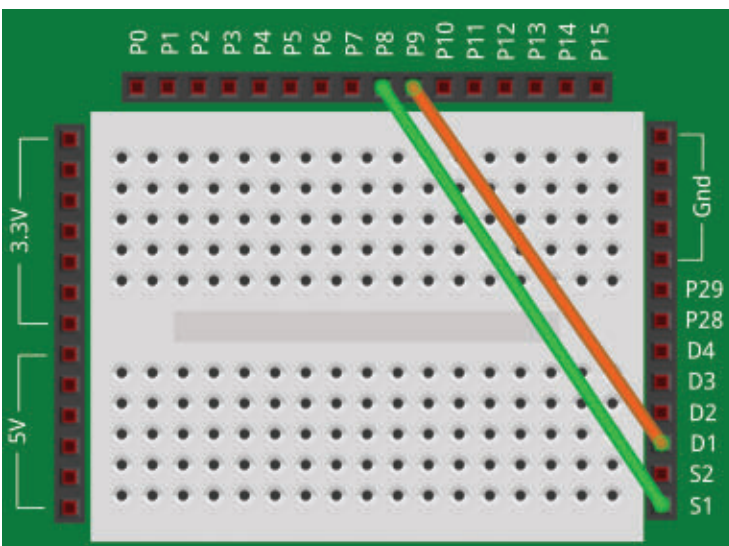


FIGURE 2: The setup for BLINK.BAS, showing input (green) and output (orange) wires.

Once you run the code, install the two wire jumpers as instructed, then press switch S1. LED D1 should blink while you hold down the button, and stop when you release it. If not, check your wiring. It's easy to misalign jumpers on the headers if you're not careful. Once you get a blinking LED, try moving the "input" wire (green in my setup in **Figure 1**) to S2 and note the change. Next move the "output" wire (orange in **Figure 1**) to D2 through D4 to see what happens. Then, change the blink

rate in lines 110 and 140 and note the difference in how your software defined switch performs. (Yes, I did miss line 120 when I renumbered this code for publication. Oops!)

This experiment should give you a good sense of the hardware available onboard your Experimenter's Section, and it opens the door to a world of simple applications that sense and control the world outside your Amigo. Let's look at what's going on here. Then, we'll move this hardware from the circuit board to the breadboard, which you can use as the foundation for your own off-board projects. We'll need to examine both the software and hardware involved in our BLINK.BAS setup (**Figure 2**).

First, let's discuss the software. Compared to other purpose-built languages for microcontrollers, Color BASIC has a limited repertoire of commands for direct I/O pin sensing and control – only two to be exact. However, these are all you need for a myriad of computer-controlled hardware applications. These two commands are *INA[x]* and *OUTA[x]*, where x is the Propeller I/O pin being used for input or output.

INA[x] reads the voltage level at pin x and converts it to either a 1 or 0 value in Color BASIC. Any voltage on pin x below a nominal 1.6 VDC returns a 0 value with *INA[x]*, and any voltage above 1.6 VDC returns a 1. *OUTA[x]* does the operation the other way around, converting a 1 or 0 value in your program to a voltage on pin x, nominally 3.3 VDC for a logic 1 and 0 VDC for a logic 0.

If you have a multimeter, you can verify this functionality by measuring the voltage on a pin, then using *OUTA* to change the state of that pin. Note that the commands *INA[x]* and *OUTA[x]* use brackets to enclose the pin number, and not parentheses or braces.

So, the command *OUTA[9]=1* in line 100 of BLINK.BAS places 3.3 VDC on pin 9, and *OUTA[9]=0* in line 130 grounds pin 9 to 0 VDC. Likewise, *INA[8]* in line 70 returns a logic 1 if the voltage on pin 8 is 3.3 VDC, and a logic 0 if it is 0 VDC.

By using these commands with some simple hardware circuits, we can enable our Color BASIC programs to "see" and "act" beyond the Amigo screen and keyboard by sensing or controlling voltage levels in those circuits. This is a pretty powerful capability!

Now, let's take a look at the hardware. **Figure 3** shows the schematic of the Experimenter's Section of your Amigo. Pin header J7 provides a modest supply of 3.3 VDC and 5 VDC for your breadboard – enough to support most simple breadboard experiments. J8 gives you access to Propeller I/O pins P0-P15. J9 provides access to ground, two tact switches (S1 and S2, to the left of the

breadboard), and four LEDs (D1 through D4, to the right of the breadboard). Also included is access to Propeller I/O pins 28 and 29, which we'll discuss in a later article.

Figure 4 shows a close-up of the Amigo circuits for the switch and LED used in BLINK.BAS. Once we understand what's going on here, we'll duplicate these two circuits to the Amigo breadboard, and then discuss how you can extend them to your own off-board creations. Let's examine these simple circuits, starting with the switch (**Figure 4**).

One terminal of the switch is connected directly to 3.3 VDC, with the other connected to ground (0 VDC) through a 4.7K pull-down resistor. The I/O pin is connected between the second switch terminal and the 4.7K resistor. This configuration places 0 VDC on the I/O pin until the button is pressed, which closes the switch and shorts the I/O pin to 3.3 VDC. The pull-down resistor prevents a direct short between the 3.3 VDC supply and ground, and it is also required to keep the I/O pin from "floating," which can lead to erroneous and unreliable results when reading the pin input state.

This configuration of a normally open switch and pull-down resistor "makes sense" to me because it returns a logic 1 ("On") via `INA[x]` when the button is pushed, and a logic 0 ("Off") otherwise. Other combinations will work equally as well. Swapping the resistor and switch in this circuit leads to a "pull-up" combination, which returns a logic 0 when the button is pushed, and a logic 1 otherwise. Also, changing the normally open switch to a normally closed one will reverse the logic levels returned in either configuration. There are some subtleties involved here, so start with the normally open pull-down configuration shown, but don't be afraid to experiment (**Figure 5**).

Here's an important point. The switch configuration in **Figure 4** is appropriate only if you're using the Amigo 3.3 VDC as the voltage source. Higher voltage sources (like 5,

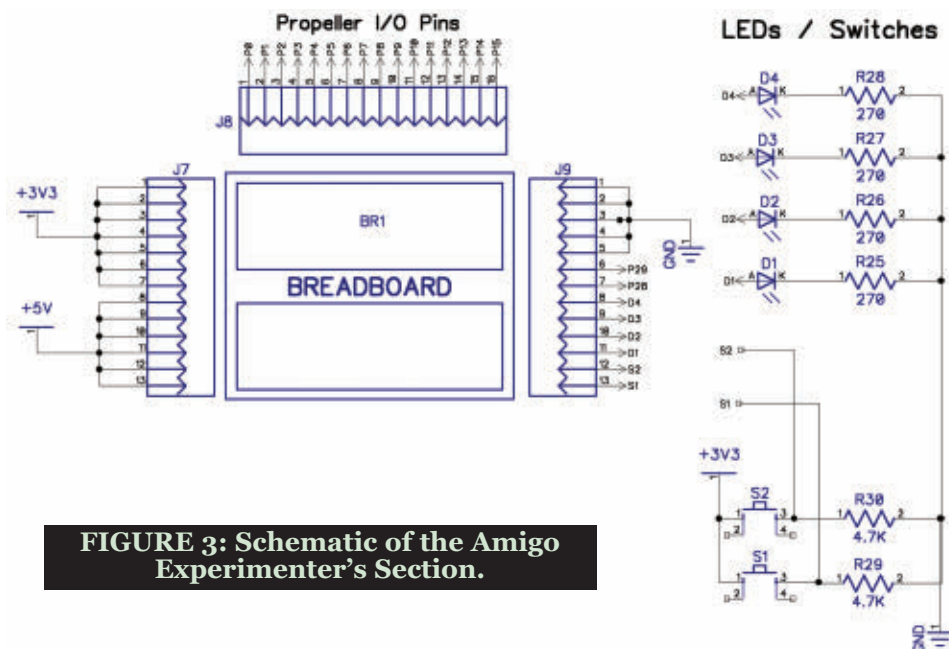


FIGURE 3: Schematic of the Amigo Experimenter's Section.

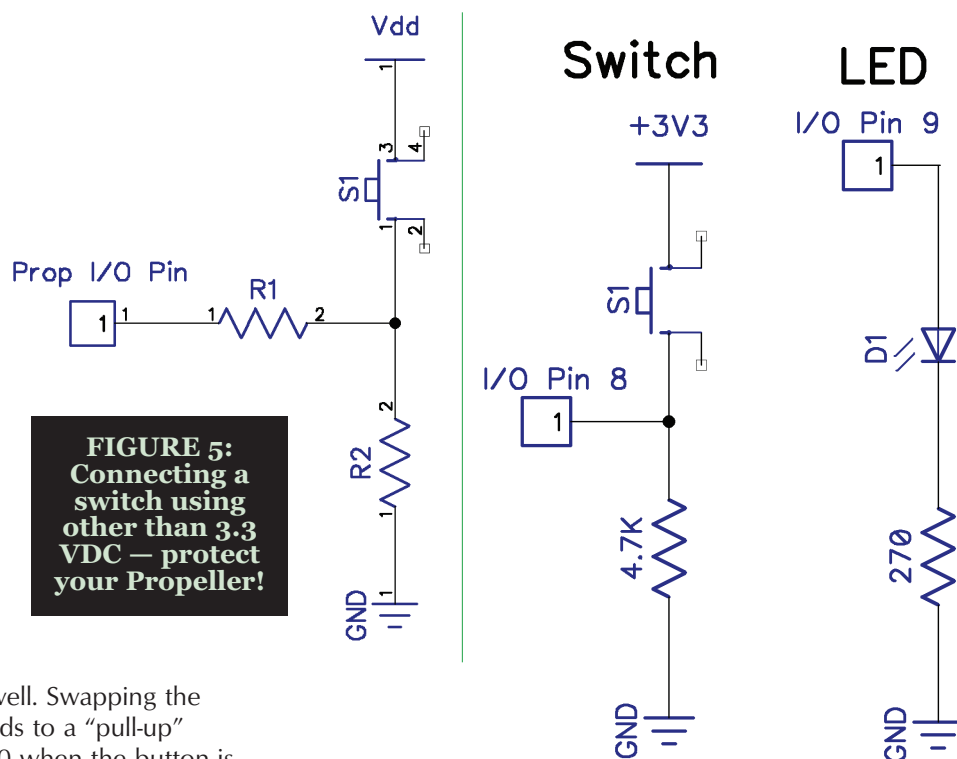


FIGURE 4: Amigo circuits for the Experimenter's switches and LEDs.

9, or 12 VDC) require the use of a current-limiting resistor between the switch and the I/O pin to protect the Propeller chip circuitry. This is because higher voltages will cause the internal protection diodes on the I/O pin to conduct, and these diodes are rated at 0.5 mA of current. You can exceed this 0.5 mA rating a bit, but applying 5

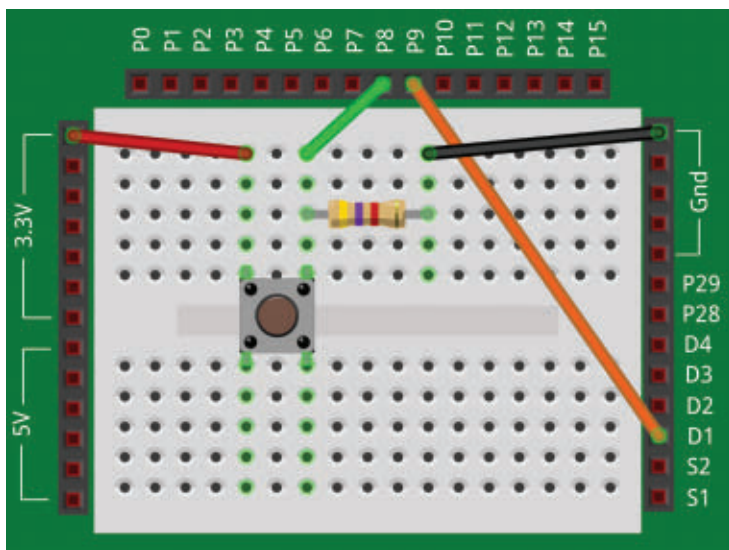


FIGURE 6: Adding an external switch with a 4.7K pull-down resistor.

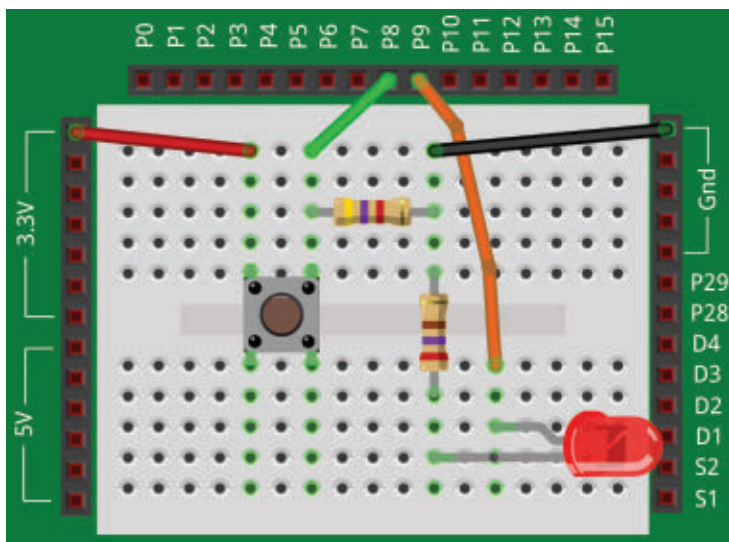


FIGURE 7: Adding an external LED and 270 ohm current-limiting resistor.

VDC (or higher voltages) directly to the I/O pin can damage your Propeller.

Use the formula for R1 in **Figure 5** to determine the value needed to limit the current through the protection diode to the rated 0.5 mA. For a switch connected to 5 VDC, this would be $R1 = 2 * (5 - 3.9) = 2.2$ kilohms. The value for R2 is less critical, and just needs to be enough to preclude excessive current drain on your supply when the switch is closed. The formula for R2 in **Figure 5** will limit the current through that branch of the circuit to 0.5 mA — a negligible level for most supplies. Again, in the case of a Vdd of 5 VDC, $R2 = 2 * 5 = 10$ kilohms.

Now, let's duplicate the Amigo switch circuit on the

breadboard. **Figure 6** shows one way to do this, using our setup for BLINK.BAS as the starting point. Begin by placing a tact switch on the breadboard; then, connect a 4.7K resistor to one switch terminal as shown. (If you don't have a 4.7K resistor handy, 10K should work just fine.) Now, wire the other switch terminal to 3.3 VDC (red wire in **Figure 5**) and the other end of the resistor to ground (black wire in **Figure 5**). Remember, we are using the 3.3 VDC supply (NOT 5 VDC!) so we don't need the current-limiting resistor between the switch and the I/O pin.

Finally, disconnect the green "input" wire from J9-S1 and reconnect it to the switch-resistor junction as shown. Now, with BLINK.BAS running, when you press the breadboard switch, LED D1 should blink. (If it blinks regardless of the button state, try rotating the tact switch 90 degrees on the breadboard.) Success!

Now, let's look at the LED circuit. The configuration in **Figure 4** is simple and straightforward, consisting of an LED connected to ground through a current-limiting resistor. The I/O pin is connected to the other lead of the LED (the anode), and the voltage on the pin determines whether the LED is on or off. Like the switch configuration, this LED configuration "makes sense" to me because the LED turns "on" when a logic 1 (3.3 VDC) is applied to the I/O pin, and it turns "off" with a logic 0 (0 VDC). Also, like the switch, this logic can be reversed by connecting the LED anode to the 3.3 VDC supply (instead of the I/O pin) and the resistor to the I/O pin (instead of ground).

In this case, a logic 0 on the I/O pin lights the LED, and a logic 1 turns it off. Again, there are some subtleties here, so start with the "makes sense" configuration and experiment with the other as needed.

Let's duplicate the Amigo LED circuit in **Figure 4** on the breadboard. As shown in **Figure 7**, add a 270 ohm resistor to the switch setup of **Figure 6**, connect the LED as shown, then move the orange ("output") wire from the J9-D1 to the breadboard. The longer lead of the LED (the anode) is connected to the I/O pin via the orange wire. The shorter lead (which will be on the flat side of the LED package, if it has one) is connected to the resistor, which is then connected to ground.

Depending on the LED, any value for the resistor in the neighborhood of 270 ohms (like 220 or 330 ohms) should keep the current through the LED to "non-smoke" levels. Now, with BLINK.BAS running, a press on the breadboard switch should set the breadboard LED flashing. (Check the LED polarity if it doesn't.) More success!

You can extend the switch to the world beyond your Amigo circuit board simply by connecting a pair of wires from the switch terminals on the breadboard to a switch (or combination of switches) across the room. This will work for the LED, as well. I've used various types of wire

for this, from store-bought to cheap audio cables to Cat 5 scraps — almost anything that's not too stiff or too difficult to solder.

To connect to the breadboard, you can solder one end of the wires to the male header pins and then cover each one with a little heat shrink tubing. Solder the other end of the wires to some alligator clips, and you've got the basis for some remote sensing and control from your Amigo.

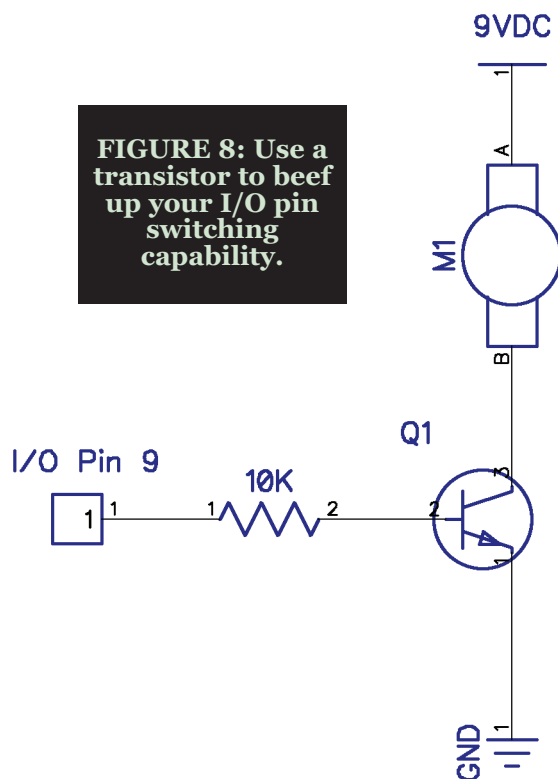
Just remember that the wire impedance and Prop I/O pin characteristics will constrain how long your wire run can be. I find six to eight feet is enough separation between most homebrew contraptions and my Amigo. Once you've completed the setup in **Figure 7**, you should be able to flash the blinking LED on the breadboard with your breadboard switch, or with a remote switch you have wired off-board with a cable and some alligator clips. Also, you should be able to remote the LED a few feet off-board by using your alligator clip cable (and paying attention to the LED polarity).

Each I/O pin of your Propeller can source or sink about 40 mA of current, with the caveat that total chip power dissipation should not exceed about one watt. This is enough to drive a couple of LEDs in parallel from a single pin, but suppose you want to control a small motor, or have some other task that has beefier power requirements.

The simplest way to extend the power handling capability of an I/O pin is to add a transistor switch. **Figure 8** shows one basic setup which uses an NPN transistor appropriate to the load and a bias resistor to protect the transistor and the Prop chip. I often use a TIP120 NPN Darlington transistor which can handle up to five amps — enough to control most hobbyist DC motors.

Several things in **Figure 8** deserve comment. First — depending on the power supply you're using for your Amigo — you'll likely find that you need a separate supply for motors and other higher current loads. I use

FIGURE 8: Use a transistor to beef up your I/O pin switching capability.



a little 9 VDC/500 mA wall wart to power my Amigo, and it works just fine — until I try to drive a little motor from the Amigo 5 VDC supply. Then, the current draw of the motor causes a brown-out reset on my Amigo, and everything starts over. So, for experiments or projects using anything but modest current loads, plan on including a separate power supply.

Next point: Don't forget the bias resistor between the I/O pin and transistor base. I find that 10K ohms work well with the TIP120 and my Amigo. Finally, if you're controlling a motor or other inductive load, use a snubber diode (the 1N4001 in **Figure 8**) to protect the transistor from the back EMF generated with you switch the motor off.

Figure 9 shows the circuit in **Figure 8** implemented on my

Amigo. I used a small DC motor which pulled about 200 milliamps, with a nine volt battery as an external supply. If you use the TIP120 as I did, remember that its pin configuration is base-collector-emitter (BCE). If you use a transistor with the EBC configuration, your breadboard setup will be slightly different.

The green and orange wires in **Figure 9** are the same ones used for input and output with BLINK.BAS; so, with that program running, pressing S1 on your Amigo causes the motor to pulse on and off in quarter second intervals.

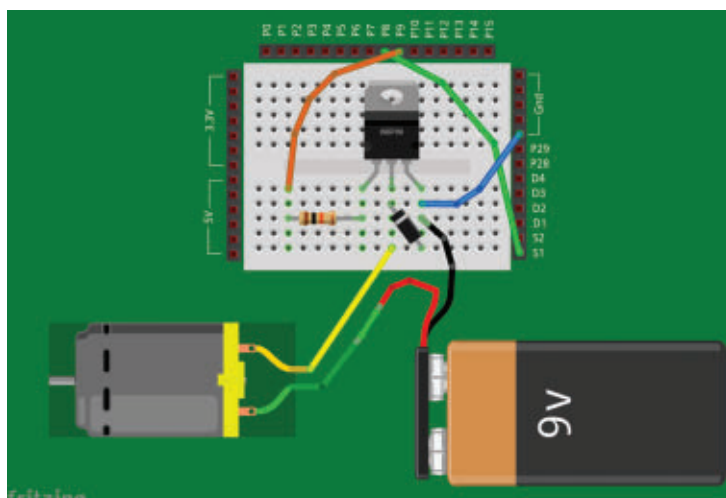


FIGURE 9: Using BLINK.BAS to pulse a motor.

Dane Weston's book on *Color BASIC for the Pocket Mini Computer* is available at www.lulu.com/shop/display-product.ep?pid=4536358.

The blue wire just creates a common ground between the Amigo and the nine volt battery.

Adiós Amigo(s)

That completes this introduction about extending your Amigo to the world outside the circuit board. With two Color BASIC commands and some simple hardware, you can greatly expand the fun (and learning!) you and your

student can have with your Amigo.

Some things to remember:

$A=INA[x]$ converts the voltage level on pin x to a 1 or 0 logic level in variable A. $OUTA[x]=1$ places 3.3 VDC on pin x; $OUTA[x]=0$ grounds pin x to 0 VDC.

Always use a current-limiting resistor between the I/O pin and any circuit using a voltage source other than the 3.3 VDC from your Amigo.

When using a transistor as a computer-controlled

switch, don't forget the bias resistor between the transistor and the I/O pin. If you don't have the specs for your transistor, check the Parallax forums, or do a quick Internet search for images associated with "I/O pin <your transistor here>."

Consider a separate power source for motors or other beefy computer-controlled loads. Don't forget the snubber diode if you're controlling a motor!

Of the 16 I/O pins brought out to J8, P0-P4 are used to interface the Prop chip with the Amigo SRAM and SD card, but are still available for use after considering that constraint. Pins P5-P15 have no assigned duties and are exclusively available for your use with the Experimenter's Section.

I hope these simple computer-controlled hardware circuits will bring you some smiles, and will fire up your imagination for cool things you can do together with a student and the Amigo. Future *Nuts & Volts* articles will discuss how your Amigo can measure resistance, hack an RC toy car, and some other fun hardware projects. Until then, be well! **NV**

Perhaps the easiest way to build your own Amigo is from the kit available from the *Nuts & Volts* webstore which provides everything you need, including the circuit board, all components (including an EEPROM pre-loaded with Color BASIC), and a 2 GB SD card with sample programs. However, if you already have a Propeller board with connectors for a PS/2 keyboard, VGA monitor, and 2 GB SD card, you can "roll your own" Amigo by downloading the Color BASIC source from the article link, updating the I/O pin assignments to match your hardware, and then loading the compiled Color BASIC binary to the EEPROM on your board.

ALL ELECTRONICS

www.allelectronics.com Order Toll Free 1-800-826-5432

INFRARED SENSOR

RE 200B is a passive infrared sensor designed to pick up heat radiation in the infrared range. Used in alarm systems and automatic switches that respond to human body motion. TO5 metal case w/ 3 leads.

CAT# PIR-200B **\$1⁰⁰** each **10 for 90¢ each**



5" FLUSH CUTTER

Thin cutting edge for hard to reach areas. Useful for jewelry, fishing & electronics. Insulated handle. Spring return.

CAT# FC-14 **\$4⁵⁰** each



12 VDC 5MM LEDs

Bright 5mm round LEDs with built-in resistors for 12Vdc. No external resistor required. Works well on 4-12Vdc. Dimmer at lower voltages.

Color	LENS COLOR	CAT#
RED	Red	LED-12R
GREEN	Green	LED-12G
WHITE	Milky-white	LED-12W
WHITE *	Water-clear	LED-12WP
BLUE	Milky-white	LED-12BW
BLUE	Blue	LED-12B
BLUE *	Water-clear	LED-12BP

* ultrabright

50¢ each

10 for 45¢ each
100 for 40¢ each



TEMPERATURE-CONTROLLED SOLDER STATION, 50W

Velleman # VTSS5U. Adjustable, 175-480°C. Protective iron rest. Tip sponge. LED power indicator. Includes pointed tip. Other tips available. CE, RoHS.

CAT# IR-50 **\$29⁹⁵** each



12 POSITION TERMINAL BLOCKS

Flexible white Nylon body, rated 105°C. 400V. Metal parts

are completely recessed to prevent shorts. Mounting hole between each circuit. Can be shortened w/ knife or saw. UL, CSA, VDE.

7 Amps **\$2⁰⁰** each **10 / \$1.80 ea.**
CAT# TB-7 **100 / \$1.60 ea.**

20 Amps **\$2²⁵** each **10 / \$2.10 ea.**
CAT# TB-20 **100 / \$1.80 ea.**



BATTERY HOLDER FOR TWO AA CELLS

Unbreakable black plastic. 6" wire leads. **CAT# BH-32**

85¢ each

10 for 75¢ each
100 for 60¢ each



170 LUMEN WORK LIGHT

Nebo 6327 LarryC. EXTREMELY BRIGHT! 170 lumens of intense light. 180° rotating magnetic clip. Water and impact-resistant. Powered by 3 AAA batteries (included).

CAT# FL-170

\$8⁹⁹ each



2.1MM COAX POWER JACK,

Panel mount. 2.1 mm center pin. Normally closed shunt opens when plug is inserted.

\$1⁷⁵ each

10 for \$1.60 each
100 for \$1.45 each



Hail, the Lowly Substitution Box!

By Robert Reed

During the '50s, '60s, '70s, and well into the '80s, capacitor and resistor substitution boxes were very popular pieces of test equipment (TE). Almost every repair shop and research lab had a collection of these devices. Now that we are in the “digital” age, they seem to have lost favor as a desired piece of test equipment. You just don't hear much mention of them anymore. I'm not sure why this came about, other than the fact that they are such simple devices. Folks may have the feeling they can easily do without them.



I acquired several of these boxes years ago, and still use them frequently today. They can be real time savers for things such as “homing” in on exact values beyond the initial design stage calculations, and for unknown or uncertain values that you need to plug into an existing design. They also can be very useful for troubleshooting procedures. Last but not least, they are very quick and easy to use.

Back when these boxes were quite popular, they were very reasonably priced and affordable by anyone — including hobbyists. Today, it's a different story as even a basic box that has decent accuracy, range, and voltage ratings are too expensive to justify their cost. A box such as this might start at \$300 or \$400. I have seen some boxes from China on eBay for well under \$100 with some decent specs. However, I did have the opportunity to see the internals of some and it left me a bit suspicious about them.

For starters, the switching was done by all plastic miniature BCD rotary switches which were PCB (printed circuit board) mounted only (no shaft to panel attachment). These boxes take a lot of hard switching over their lifetime, so this leaves me doubtful about their mechanical longevity. I cannot speak for all of them; only the ones I have had the opportunity to tear down. SparkFun Electronics has a starter kit (resistive) that has nice specs, and from the pictures they show it looks to be of decent construction and at a very low price.

Start working your way up in tighter specs, and substitution boxes can run over \$1,000.

The very top of the heap can go for upwards of \$30,000! Of course, these boxes have specs that would rival primary standards, and are way beyond our needs or even our capabilities to make use of them. By contrast, the boxes I purchased years ago are five decades of value, 1% accuracy, and 600 volt/one watt ratings, and cost well

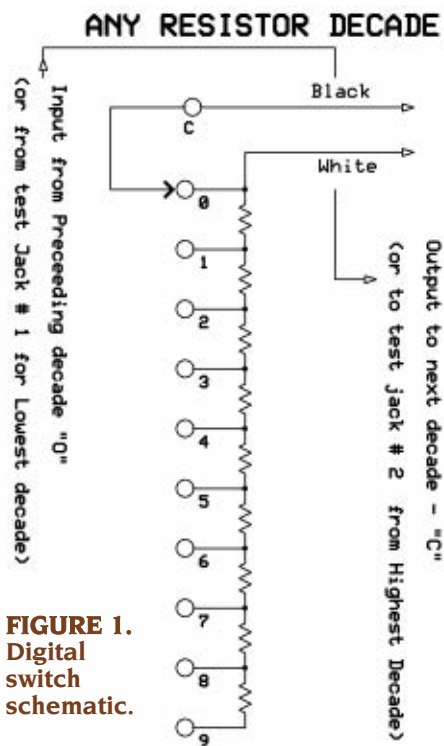


FIGURE 1.
Digital
switch
schematic.



FIGURE 2.

below \$100 in today's dollars. Plus, they all still meet factory specs.

Types of Boxes

Resistor and capacitor boxes come in two basic styles — one being a hit-and-miss selection of standard values, which are the cheapest and least useful. The other is in decade values which will cover four, five, or six orders of magnitude in range, with a resolution of the lowest decade range (i.e., lowest range of 100 pF — steps in 100 pF to the largest value of the box — and lowest range of 10 ohms — steps of 10 ohms to the largest value of the box).

Of these types, the only one I would consider is the decade style due to its wide range and almost infinite resolution (based on the lowest decade range). Just imagine that a seven decade box can cough up 10 million values of resistance in one ohm steps!

Substitution boxes are still being manufactured and are available through most retailers. They come in a variety of electrical switching schemes such as digital, rotary, and slide switches. The thumbwheel switches are of modest cost, and will be either in straight decimal or BCD format. (The decimal type requires nine component values per switch, while the BCD type will only require four component values.) Each switch covers one decade of values (0-9) that are stacked side by side. They will read out in numerical values for each segment. When all are set, they will give a direct readout of the total value such

as 0.241200 μ F or 520,970 ohms.

However, their voltage ratings are 50 VDC or at best maybe 100 VDC, and low wattage.

This limitation may be of no concern to you, but if it is and you desire higher voltage and/or wattage, you definitely have to go with the rotary style switching or slide switches.

When using the rotary switch type to accommodate higher wattage/voltage components, it can become very pricey as the voltage and value ratings increase. Each switch will cover 11 positions: a full decade plus zero (0-10). This type offers good ergonomics and speed of switching. Then, comes the slide switch style; these are the lowest cost version of the switching schemes.

This configuration will have four individual switches per decade: either in a 1-2-3-4 sequence, or a BCD format with a 1-2-4-8 sequence. The panel is usually arranged as four vertical switches which covers one decade; the value is

numerically labeled as the numbers just explained. Then, each decade just repeats in a horizontal direction.

So, if it has six decades of range, it will require 24 slide switches (four values times six ranges) with each decade labeled in the same sequence, but with higher multipliers. By adding up all the switch values that are engaged, you come up with a total value.

These switches have the advantage of high voltage and a cheap selling price. However, after using one of these on several occasions, it almost drove me nuts trying to keep track of adding and subtracting numbers constantly (1-4 per decade) to obtain the actual value. They also require a lot of panel machining for rectangular slots and mounting holes.

The Internet has a lot of info on constructing these boxes but — as usual — it tends to be hit-skip information without much mention of advantages/disadvantages about their operation and usefulness. What I want to describe in this article is various types of DIY construction of all the various switching modes, along with the benefits and caveats of each type. I will only cover the basics of construction and leave it up to the builder as to enclosures, knobs, etc., to suit his/her fancy.

The Resistance Decade Box

Let's begin with resistor sub boxes since these will be by far the simplest and cheapest. We will look at this in three different switching schemes: digital, rotary, and slide switches.

From Preceding Decade Wiper
(or test Jack #1 for lowest decade)

ANY RESISTOR DECADE

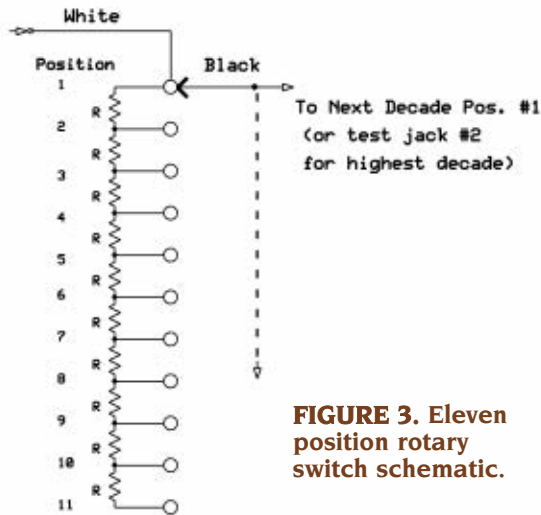


FIGURE 3. Eleven position rotary switch schematic.

Starting with the digital switch, the type needed here is a decimal version (not BCD). Refer to **Figures 1 and 2** for a wiring diagram and a completed switch with components installed for any one decade. Note that physical pinout locations may vary according to manufacturer.

The pin arrangement of the one I used is fairly common, in a 9-8-7-6-5-C-4-3-2-1-0 from one side of the board to the other. Each switch section will output a 0-9 value of its particular decade value. Then, the switch sections are stacked and interlocked side-by-side to complete the unit. You can go as high or low in decade units as desired, but I have found that six decades in 10s, 100s, 1Ks, 10Ks, 100Ks, and 1Ms should be quite sufficient.

I have very rarely needed any value outside of that range, and inherent wiring and contact resistance degrade the 1s decade. However, a 1s decade does have some merit in that the overall readout would be in total ohms and eliminate the need to mentally add a zero after the last digit. Alternatively, one could just add a “dummy” section in that slot and lock it down to “0” (assuming the succeeding decade was 10s).

Most digital switches have an upper limit of 50 VDC or at most 100 volts, and lower current ratings to boot. If that will meet your needs, then you can use 1/4W resistors or possibly 1/2W on the 10s or 1s decade to accommodate higher currents on these ranges (although space is at a premium here). To make this unit ultra small, SMD components really shine here. One percent SMDs are not hard to find, but as they go up in wattage rating, the job gets a little tougher.

As to accuracy, you can get by with 5% tolerances if you’re not too fussy. In my experience using 5% carbon film, their tolerance is almost always within 2-3%.

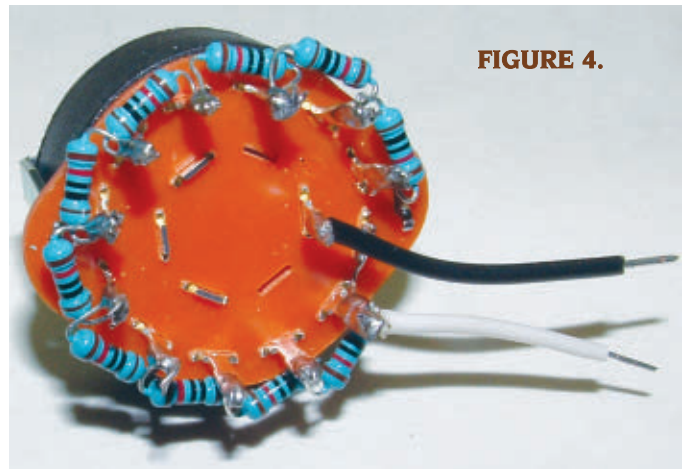


FIGURE 4.

Fortunately, 1% is not priced much more than 5%. You will need nine resistors per decade, so you will be looking at 45-63 components and the difference in price between the two might be \$3 at the most.

The upside of this style of construction is very compact boxes, a digital display of the total resistance dialed up, and a low parts cost. The downside is (as mentioned) limits of voltage and power ratings. I used 1/4W/1% resistors in this one, and the populated switch shown cost me \$0.82 per decade. Add \$5 for a box and the total cost was about \$10.

Next up is the rotary switch style. Refer to **Figures 3 and 4** for a wiring diagram and a completed switch with components installed for one decade. The desirable contact arrangement here would be single pole/11 position. This would allow for 0-10 values per decade, and it is convenient to have that extra value (10) when in actual use. However, most of these switches will be 12 position with an occasional 10 position.

If 10 position is the only one available to you, then the arrangement can be 0-9 values per decade. With 12 positions, you could do the switching sequence in a 0-10-0 value. The zero on each end of the switch is not totally undesirable, and occasionally speeds things up when rapid firing through the decade ranges. Or, you could just bury that extra position with double zeros at the onset of switching.

A word of note here is that some rotary switches come with a removable position stop allowing you to limit the maximum position to whatever you want. If you have a choice, get the “make-before-break” type of switching. This will ensure that the circuit area you are using it on will never see a momentary open condition when switching in the different resistor values.

However, opening “live” circuits momentarily in most cases will not have much effect on them, but some thought will be required for the consequences of that stage and the circuitry downstream from it. For those occasions where you are in doubt, you can momentarily

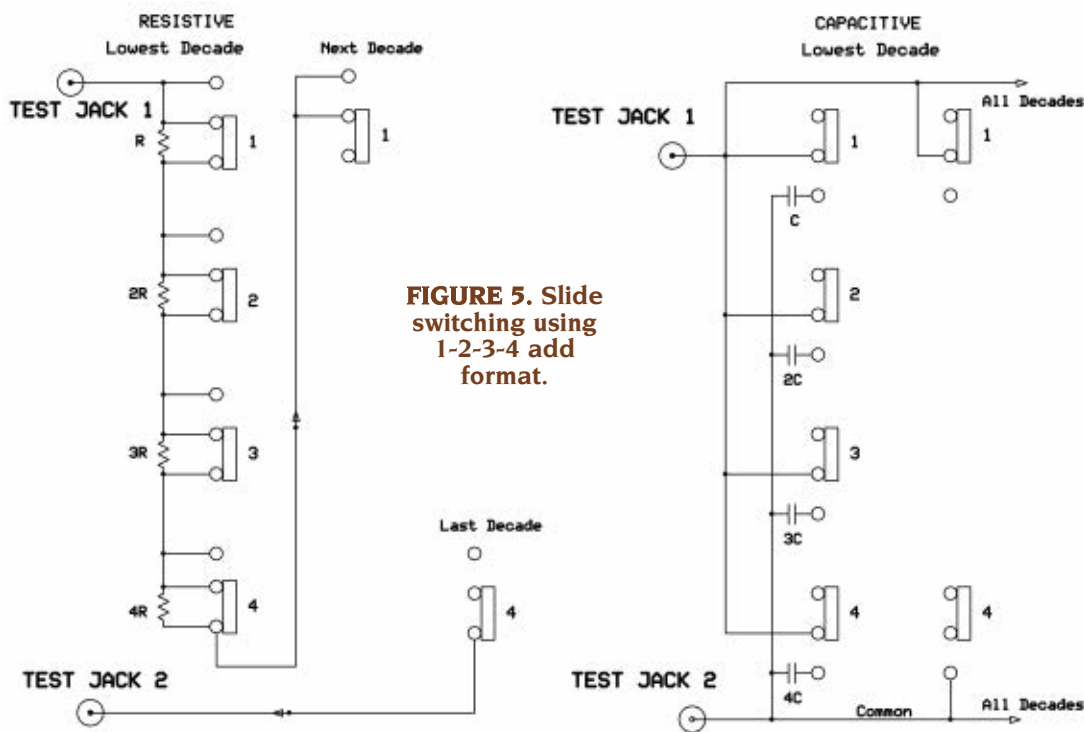


FIGURE 5. Slide switching using 1-2-3-4 add format.

	BCD Switch Terminals Connected to "C"			
	1	2	4	8
0				
1	■			
2		■		
3	■	■		
4			■	
5	■		■	
6		■	■	
7	■	■	■	
8				■
9	■			■

ANY DECADE SWITCH

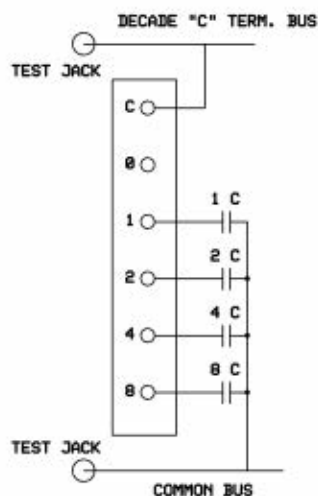


FIGURE 6. Capacitance box digital switch.

unit, adding an extra measure of safety for those "OOPS" moments. Also, they are faster and easier to switch. In this style, you will definitely want to go with 1/2W/1% resistors, maybe upping that to 1W on the 1s, 10s, and possibly the 100s ranges.

The downside is a bulkier and somewhat more expensive unit than the digital style.

I have seen rotary switches priced at less than \$3 from the major suppliers to as low as less than \$1 from eBay and Tayda Electronics (and they were make-before-break style). Having used all three switching styles presented here, the rotary version is my first choice by far for the reasons already stated and are just a pleasure to use.

I'm not going to dwell much on the slide switch type. Construction is simple and straightforward, and a schematic is included in **Figure 5**. The upside on this style is it's very economical to build for high voltage and high capacity operation. The downside is a lot of panel fabrication is involved and it's a real pain to operate.

The Capacitance Decade Box

Next up is the capacitor decade box. These will be more expensive to build. Again, starting with the digital type of switching, most advantages/disadvantages apply here as its resistive counterpart. For the digital style of switching, the BCD type is the preferred switching mode here — especially since all caps are connected in parallel to a common bus. This will use four base values of capacitance in a 1C, 2C, 4C, 8C arrangement.

could not pass up the price. This is a make-before-break switch and has no stop, which is kind of neat since you can run through the positions in either direction no matter what location it is on. The cost per populated decade switch averaged \$1.30. Total cost for six decades, box, and knobs was about \$18. The completed unit is shown in **Figure 10**.

The upside of rotary switching is higher voltage/wattage specs can be extended over the digital type making for a more rugged and versatile

power-down between switching in different values.

In the switch shown, I used only 1/4W resistors for better illustration. In the actual completed unit, I used 1/2W on the upper ranges and 1W on the lower ranges — all 1%. I also went with a 12 position switch because I

It will need two caps in parallel for three of these values because base values 2, 4, and 8 are almost impossible to obtain, so – in reality – it will take seven caps per switch (i.e., $0.47\ \mu\text{F} + 0.33\ \mu\text{F} = 0.8\ \mu\text{F}$).

Obviously, these will occupy more space than the pinout board can handle, so they will partially hang off the pinout board and require a little more box depth than the companion resistor box did. Since these switches have a 50V limit, SMD capacitors would work nicely here if you can find the tight tolerance you desire at an affordable price.

The SMD parts will also allow for a smaller box. Refer to **Figures 6 and 7** for the schematic diagram and an image of a completed switch using through-hole components.

As opposed to the series connected components in the resistor boxes, capacitor box components are all wired up in parallel to a common bus wire. The base value components are each connected to their corresponding pinout numbers (base value 1 - pin 1; base value 2 - pin 2, etc.). The C terminal is then tied in parallel to the other decade switch's C terminals and to one of the test jack terminals.

The other leads of all capacitors are connected to the common bus and to the other test jack terminal. For this switch, I used 5%/100V metallized polyester capacitors, so a populated decade switch cost me \$2.20 each. Although you can use as many active decades as desired, I only used four: one active but unpopulated decade, and two “dummy” decades to complete this box. The readout (shown in **Figure 10**) is 0.000040 microfarads.

The unpopulated leading digit is awaiting a decent price and size for these larger components. The last two digits (dummies) have been locked in place for ease of reading the total display in microfarads. Why are these two locked at “40?”

Since the parasitic capacitance of most capacitor decade boxes is around 30-50 pF, there is not much need for a 10s and 1s digit because they are so severely degraded. As long as these digits are present, they might as well be locked at 40 which is the exact total parasitic capacitance of this assembly. This way, they serve a more useful purpose.

The end result displayed is 0.xxxx40 μF , with ‘x’ identifying the active decades. In time, the leading digit will be populated for future use. As it stands now, the total cost including the box for this completed unit was \$16.

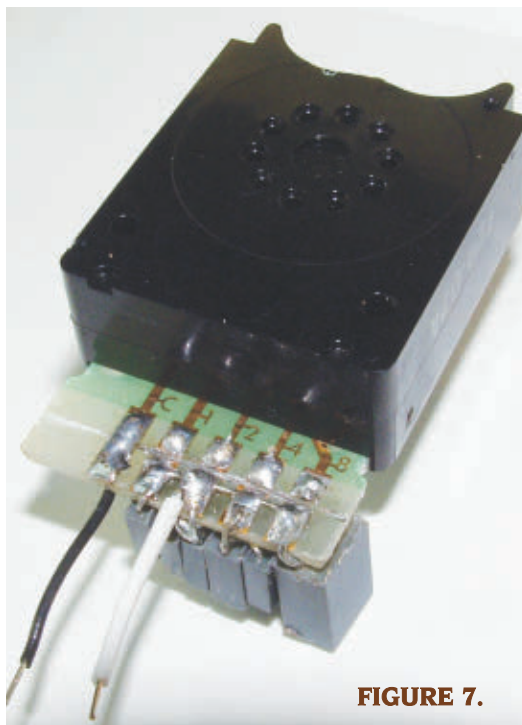


FIGURE 7.

The Capacitance Decade Box with Rotary Switching

Now for the “king of the hill” capacitive decade box: the rotary style of switching. This can get quite expensive since these switches can handle about any capacitor you want to wire up to them – high voltage and/or high value. A lot of decisions and web searching may have to be made here to reach a balance as to quality vs. cost. Depending on switch arrangement, there are only three choices here.

First, a single pole/11 position wafer switch with 10 base values (0-10) of capacitors for each decade. The switch is cheap, but the cost of capacitors would be out of sight! Next option would be a

BCD type of switching using only four base values (1-2-4-8). A big savings here on capacitors, but the switch would require four poles and would need four separate decks to accomplish this. Less cost in terms of capacity, but the cost for the bare switch from major distributors will be astronomical (\$60-\$70) and very bulky.

An interesting side note here is that the relatively cheap decade boxes of the past used a special BCD arrangement that had only one wafer. It had a couple more terminal lugs and a special cut wiper plate so that manufacturing costs were little more than today's simple and cheap single deck 11 position wafer switch. It's sad to say these are now unobtainable, at least to the best of my knowledge.

I really gave this one a lot of thought and came up with a pretty good solution. I call it a “10 of 5” switch which utilizes two wafers in a DP/11T configuration and five base value caps to produce the whole decade of 10 values. Refer to **Figures 8 and 9** for the schematic and populated switch. Again, 5%/100V metallized polyester film caps were used here. The common end of the caps was soldered to the switch's metal binding bracket on the rear as shown.

These points were ultimately wired together and tied to the common test jack after installation to the front panel. When I placed an order for these caps, I ordered three of each value because their price was so low. To you newbies out there, you may want to do the same if a fantastic deal presents itself. Not only do you amortize the postage over more parts, but this is how you eventually build up your stock. After almost 40 years of over-ordering

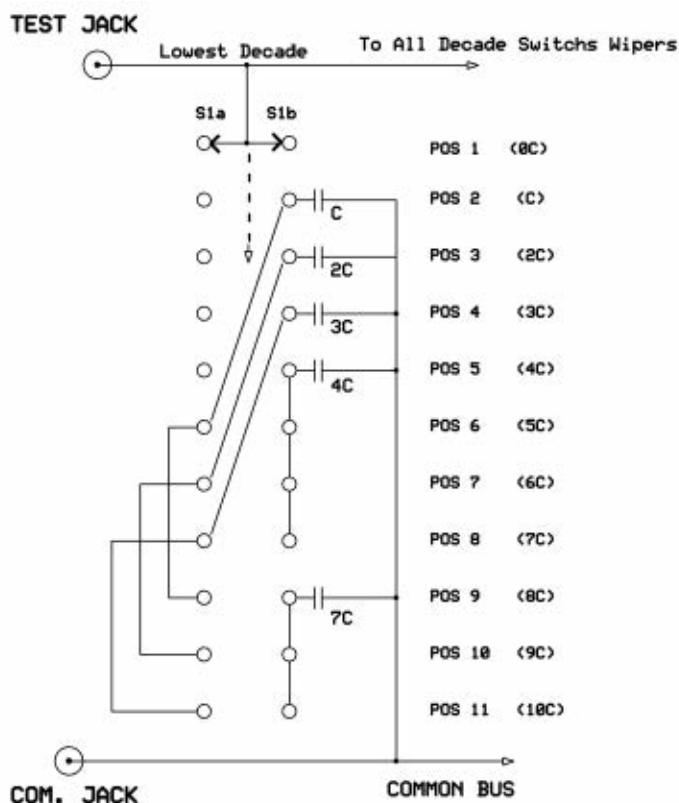


FIGURE 8. Capacitance box 11 position rotary switch.

like this, I have built up a stockpile of components that almost anything I need is within arm's reach. It will increase the pleasure of your hobby tenfold by doing this.

Before installing these caps, I got curious as to how accurate they really were and went about checking each one. More than nine out of 10 were within 2-3% of the stated value. That's pretty close to the target value and adequately sufficient for subbing. The decade switch pictured cost \$3.10 (each) and the caps averaged \$2.20

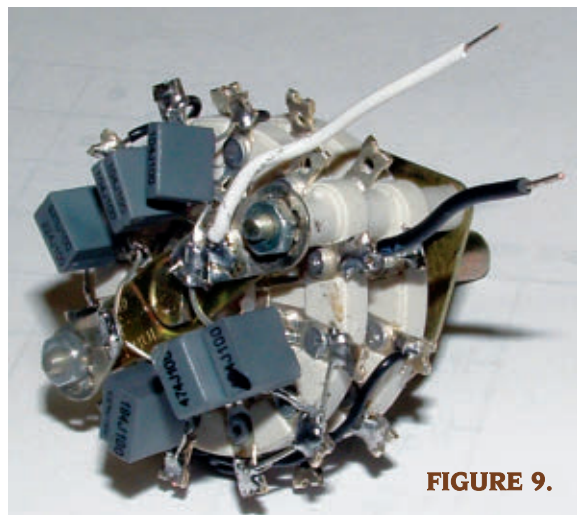


FIGURE 9.

per decade (\$5.30 for a populated switch). Adding \$10 for the box and knobs, a four decade box (100 pF to 1.0 μ F) cost \$31 total.

Of course, when using a switch like this that can handle just about anything you can attach to it, the ultimate goal would be 1% tolerance and several hundred volt ratings, and possibly one more decade of 1 μF steps. If one were to realize this, it would not come cheap, but with enough web searching and careful buying you could come close. For the highest value decade (1-10 μF), 10% tolerance would be sufficient and keep the cost down on these larger values. As the old engineering saying goes, "It's not perfect, but close enough."

Closing Notes

These sub boxes perform their best from DC to several MHz, so when used at higher frequencies, one has to take into consideration internal parasitic reactances that will have more effect as the frequency goes upward. So, basically, resistor box components are always wired in

series; capacitor box components are always wired in parallel.

Resistor boxes are very inexpensive to build. Capacitor boxes can be relatively inexpensive to build or they can be very expensive depending on the number of decades used and the voltage/tolerance level desired. This covers a lot of levels of increasing quality. The choice is yours.

Whatever you decide, I would recommend you populate the decade switches first; lay them out as they would ultimately be assembled to the front panel

**FIGURE 10.**

and then purchase a box to fit. However, with a little forethought on the sizes of switches and components, you could possibly order the box from the same supplier thereby saving on postage. The resistor box shown in **Figure 10** is 3-1/2" W x 5-1/4" L x 1-1/4" H (a little tight). The capacitor box in that same photo is 4" W x 2-1/2" L x 1-1/2" H (a little deep to allow for test jacks on the rear which is hidden from view in the image). Also, the white and black leads for decade switch connections are just to help clarify these points to their respective schematics. In practice, the decades would all be wired together with one color.

As much as I dislike suggesting secondary distributors, the cost of switches from major distributors is just too prohibitive. The resistive single pole/12 position switch I used was purchased from Tayda Electronics, along with most of the components and boxes. At the time of this writing, the switch was SKU-1637 at \$0.89 (they assured me that a stock of 500 was always kept). You can download their catalog and order online at www.taydaelectronics.com.

All the rest of the switches shown were purchased on eBay from China. There just seems to be an unending supply there from a multitude of sellers. Prices and sellers keep changing frequently, so rather than recommending one over the other, you're better off just plugging into the eBay search box and go for the best deal. (Of course, you should check out what *Nuts & Volts*' advertisers have to offer first!) The capacitive rotary switch is a two pole/11 position, and will have two wafer decks. The digital switches usually come in a bank of 10.

Two cautions here: Use straight decimals for resistors and BCD for capacitors; and check the specs for the height of the switch which should be about 1-1/4" high (most of these will be this height).

Although there is a lot of information on the Internet on sub boxes, my objective here was to consolidate all the information into one "arena" so you can weigh the styles and options available. I hope

this will simplify your choices if you decide to DIY your own.

Even with simple projects, questions sometimes arise and if so, you can email me at rjr@ncweb.com. Plus, I can send the panel artwork for a variety of different switch configurations.

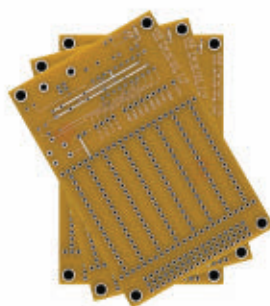
The parts for these boxes are quite inexpensive and easy to construct. Build a couple, use them, and enjoy them for years to come! **NV**

FREE

PCB Layout Software &
PCB Schematic Software



expresspcb.com

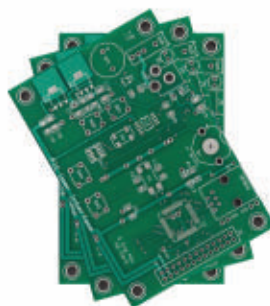


3 PCBs

\$41 + shipping

2-layer
MiniBoard

(Shipped in 1 business day)



3 PCBs

\$61 + shipping

2-layer
MiniBoardPro

(Shipped in 2 business days)



3 PCBs

\$81 + shipping

4-layer
MiniBoardPro

(Shipped in 3 business days)

- 1** **DOWNLOAD** our free CAD software
- 2** **DESIGN** your 2 or 4-layer PCB
- 3** **CHECK** your design with xCHECK DRC
- 4** **SEND** us your design with just a click
- 5** **RECEIVE** top quality boards in just days

Now Offering
**GROUND
SHIPPING**

- Continental US Customers -

LIMITED TIME PROMOTION | www.expresspcb.com

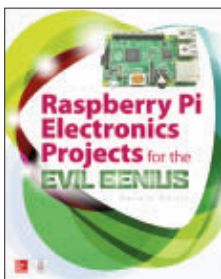
For complete product details, visit us online!!

The Nuts & Volts **WEBSTORE**

GREAT FOR DIYers!

Raspberry Pi Electronics Projects for the Evil Genius by Donald Norris

This fully illustrated TAB guide shows how to construct and program all kinds of fun and innovative gadgets with your Raspberry Pi. *Raspberry Pi Electronics Projects for the Evil Genius* features 10

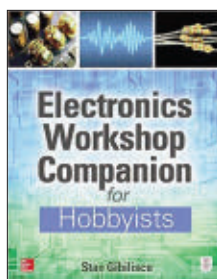


complete projects that showcase cool RasPi applications in computing, communications, robotics, photography, and video. Each inexpensive project includes a detailed list of materials, sources for parts, schematics, and clear step-by-step assembly and programming instructions.

\$25.00

Electronics Workshop Companion for Hobbyists by Stan Gibilisco

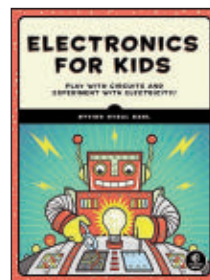
In this practical guide, electronics expert Stan Gibilisco shows you, step by step, how to set up a home workshop so you can invent, design, build, test, and repair electronic circuits and gadgets. *Electronics Workshop Companion for Hobbyists* provides tips for constructing your workbench and stocking it with the tools, components, and test equipment you'll need. Clear illustrations and interesting do-it-yourself experiments are included throughout this hands-on resource.



Reg Price \$25.00 Sale Price \$19.95

Electronics for Kids by Oyvind Nydal Dahl

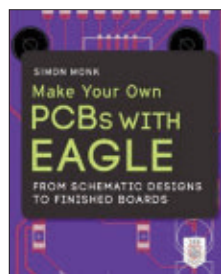
Why do the lights in a house turn on when you flip a switch? How does a remote controlled car move? And what makes lights on TVs and microwaves blink? The technology around you may seem like magic, but most of it wouldn't run without electricity. In Part 1, you'll learn how current, voltage, and circuits work. In Part 2, you'll make even more cool stuff. Then, in Part 3, you'll learn about digital electronics — things like logic gates and memory circuits work.



\$24.95

Make Your Own PCBs with EAGLE by Eric Kleinert

Featuring detailed illustrations and step-by-step instructions, *Make Your Own PCBs with EAGLE* leads you through the process of designing a schematic and transforming it into a PCB layout. You'll then move on to fabrication via the generation of standard Gerber files for submission to a PCB manufacturing service. This practical guide offers an accessible, logical way to learn EAGLE and start producing PCBs as quickly as possible.

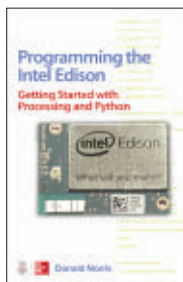


\$30.00

Programming the Intel Edison: Getting Started with Processing and Python by Donald Norris

Learn To Easily Create Robotic, IoT, and Wearable Electronic Gadgets!

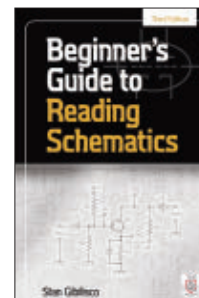
Discover how to set up components, connect your PC or Mac, build Python applications, and use USB, WiFi, and Bluetooth connections. Start-to-finish example projects include a motor controller, home temperature system, robotic car, and wearable hospital alert sensor.



\$20.00

Beginner's Guide to Reading Schematics, 3E by Stan Gibilisco

Navigate the roadmaps of simple electronic circuits and complex systems with help from an experienced engineer. With all-new art and demo circuits you can build, this hands-on, illustrated guide explains how to understand and create high-precision electronics diagrams. Find out how to identify parts and connections, decipher element ratings, and apply diagram-based information in your own projects.

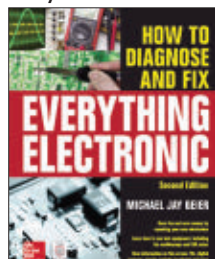


\$25.00

How to Diagnose and Fix Everything Electronic by Michael Jay Geier

A Fully Revised Guide to Electronics

Repair all kinds of electrical products, from modern digital gadgets to analog antiques, with help from this updated book. The Second Edition offers expert insights, case studies, and step-by-step instruction from a lifelong electronics guru. Discover how to assemble your workbench, use the latest test equipment, zero in on and replace dead components, and handle reassembly. **\$24.95**



Programming PICs in Basic by Chuck Hellebuyck

If you wanted to learn how to program microcontrollers, then you've found the right book! Microchip PIC microcontrollers are being designed into electronics throughout the world and none is more popular than the eight-pin version. Now the home hobbyist can create projects with these little microcontrollers using a low cost development tool called the CHIPAXE system and the Basic software language. Chuck Hellebuyck introduces how to use this development setup to build useful projects with an eight-pin PIC12F683 microcontroller. **\$14.95**



Programming Arduino Next Steps: Going Further with Sketches by Simon Monk

In this practical guide, electronics guru Simon Monk takes you under the hood of Arduino and reveals professional programming secrets. Also shows you how to use interrupts, manage memory, program for the Internet, maximize serial communications, perform digital signal processing, and much more. All of the 75+ example sketches featured in the book are available for download. **\$20.00**



Order online @ store.nutsvolts.com
Or CALL 1-800-783-4624 today!

EDUCATIONAL

Beginner's Guide Educational Combo!

Price Reduced
Now Only \$129.95

An Arduino Workshop

Are you puzzled about the Arduino but finding it difficult to get all the pieces in one place?

Joe Pardue
SmileyMicros.com

Puzzled by the Arduino?

Based on *Nuts & Volts* Smiley's Workshop, this set gives you all the pieces you need!

Educational Combo \$124.95

We now have the fully updated Learning Labs 1, 2, & 3 in stock

Are you (or someone in your family) looking to get started in engineering, electronics, computers, or robotics? If so, this hands-on comprehensive Learning Lab series is for you! Each Lab includes hands-on exercises and training on practical electronics, components, and circuits. Complete details on all the Learning Labs are available in our website. Simply type "lab" into the search field at <http://store.nutsvolts.com>!

Educator Discounts Available!

CD-ROM SPECIAL

Nuts & Volts 12 CD-ROMs & Hat Special!

That's 144 issues. Complete with supporting code and media files.

Free Shipping!

Only \$249.95 or \$24.95 each.

The Nuts & Volts Pocket Ref

All the info you need at your fingertips!

This great little book is a concise all-purpose reference featuring hundreds of tables, maps, formulas, constants & conversions. AND it still fits in your shirt pocket!

Only \$12.95

Visit store.nutsvolts.com or call (800) 783-4624

Help is Finally Here for 32-bit PICs

Finally! Microchip has given us a 32-bit PIC microcontroller that is supported like its famous eight-bit microcontrollers. Now, 32-bit embedded applications can be developed using the new series of 32-bit PIC32MM parts with the MPLAB Code Configurator, which originally only supported eight-bit PIC microcontrollers. The new PIC32MM PICs are low power/low pin count devices that carry many of the same core peripherals that are common to the PIC16F line of enhanced PICs. With the assistance of the MPLAB Code Configurator, we can quickly produce code for all of these peripherals and more.

Tools and Base Configuration

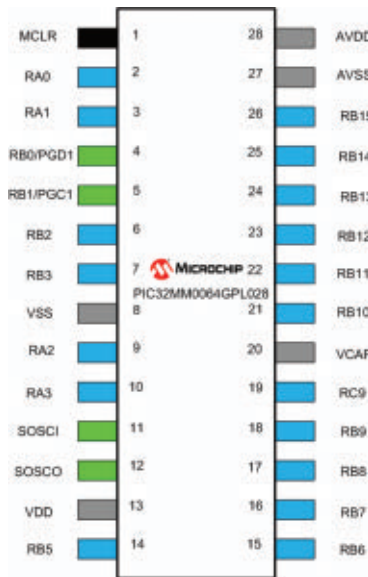
The PIC32MM tool chain is free. To follow along, you'll need to download and install the latest version of MPLAB X. You will also need to download and install the latest version of XC32. Be sure to install the MPLAB Code Configurator plugin within MPLAB X. Once you've gotten your application assembled, you can debug and program the PIC32MM by attaching a PICkit3 to your PIC32MM hardware.

We will be working with the PIC32MM0064GLP028. The PIC32MM0064GLP028 is the only variant of the PIC32MM family that can be had in a PDIP package. The PDIP package allows you to put our reference design down on a solderless breadboard if you desire.

An MPLAB Code Configurator view of the PIC32MM0064GLP028 can be seen in **Figure 1**. **Schematic 1** is rather empty as it represents a bare-bones PIC32MM0064GLP028 configuration.

We will configure the PIC32MM0064GLP028 with an external 8 MHz crystal in our hardware, as well as within the Code Configurator. This will allow us to utilize the PIC32's internal PLL to drive our system clock frequency at 24 MHz. We will also default to PGED3 and PGEC3 as the programming/debugging portal. The ICSP setup is in standard Microchip PICkit3 format.

Rather than pull



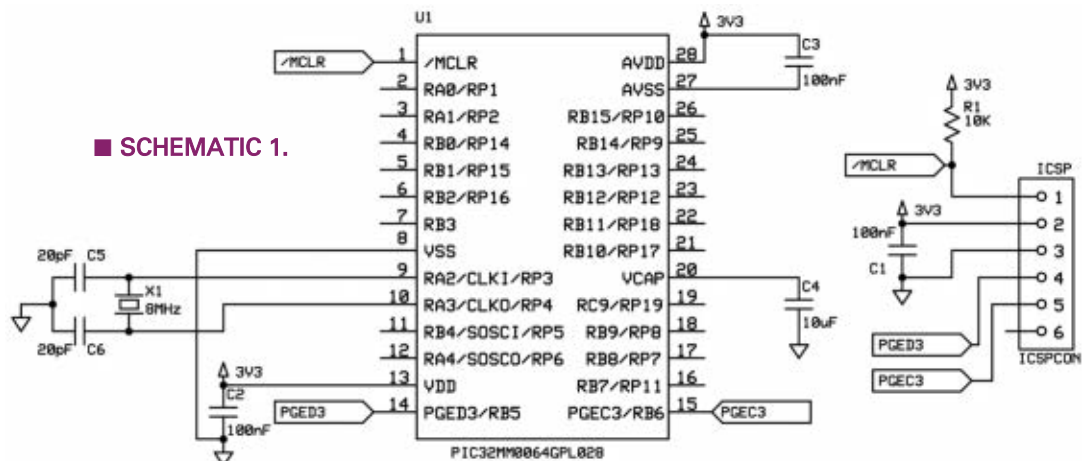
■ **FIGURE 1.** This is the MPLAB Code Configurator view of the PIC32MM0064GLP028. As we build our PIC32MM0064GLP028 application, this view will reflect our configuration changes.

PIC32MM routines out of our PIC32MX tool box, we will rely on the code produced by the MPLAB Code Configurator. We will use the Code Configurator to lay down the base code for our peripherals. This base code can then be applied as necessary to our application. With that, let's get started.

System Setup

This discussion assumes that you are familiar with MPLAB X and the MPLAB Code Configurator. If not, there are tutorials within MPLAB X and on the Microchip website that you will find useful. The first logical order of business is to set up the PIC32MM0064GLP028 configuration fuses. In the process of setting the configuration fuses, we will also assign the programming/debugging pin set and reserve the primary oscillator pins for the external 8 MHz crystal.

■ **SCHEMATIC 1.**



In **Screenshot 1**, I've clicked on the proper check boxes and made the logical selections within the MPLAB Code Configurator to reserve and assign the desired PIC32MM0064GLP028 pins. Note that I have enabled the PLL and multiplied the 8 MHz input frequency by three to clock the PIC32MM0064GLP028 at 24 MHz.

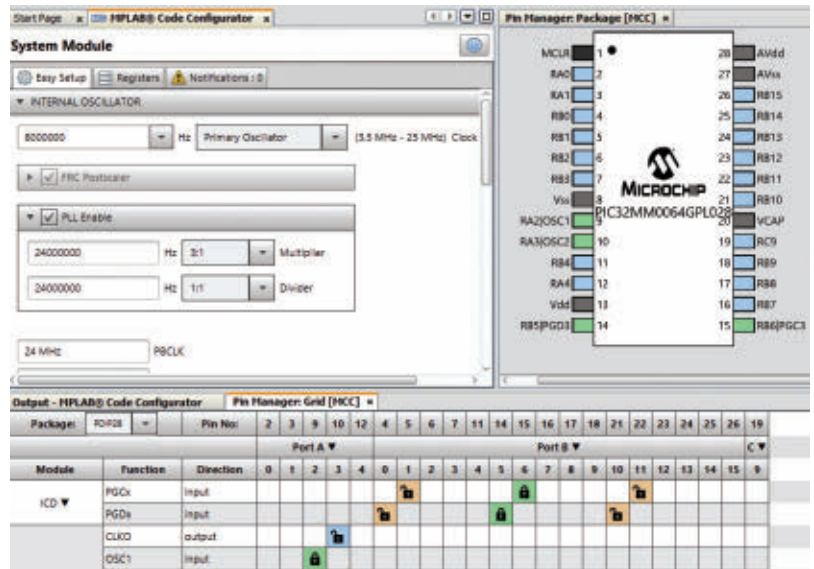
Clicking on the Code Configurator GENERATE button produces code that is directly related to the system choices we made in the MPLAB Code Configurator. This generated code is automatically placed into the correct folders of our MPLAB X project. All of the Code Configurator-generated files are placed into subfolders named *MCC Generated Files*. You can observe the placement of the Code Configurator-generated files in the *Project* view of MPLAB X under *Header Files* and *Source Files*.

At this point, we have used the MPLAB Code Configurator to create a number of files, which include *main.c*, *mcc.c*, and *mcc.h*. Code within the aforementioned files is sufficient to initialize the PIC32MM0064GLP028. Here's a look at the PIC32MM0064GLP028 configuration fuses that were generated by the MPLAB Code Configurator and placed into *mcc.c*:

```
// Configuration bits: selected in the GUI
// FDEVOT
#pragma config SOSCHP = OFF
// FICD
#pragma config JTAGEN = OFF
#pragma config ICS = PGx3
// FPOR
#pragma config BOREN = BOR3
#pragma config RETVR = OFF
#pragma config LPBOREN = ON
// FWDTP
#pragma config SWDTPS = PS1048576
#pragma config FWDTPWINSZ = PS25_0
#pragma config WINDIS = OFF
#pragma config RCLKSEL = LPRC
#pragma config FWDTEN = OFF
// FOSCSEL
#pragma config FNOSC = PLL
#pragma config PLLSRC = PRI
#pragma config SOSCHP = ON
#pragma config POSCMOD = XT
#pragma config OSCIOFNC = OFF
#pragma config SOSCHSEL = ON
#pragma config FCKSM = CSECME
// FSEC
#pragma config CP = OFF
```

The MPLAB Code Configurator-generated file *mcc.c* also contains the system initialization function:

```
void SYSTEM_Initialize(void)
{
    PIN_MANAGER_Initialize();
    OSCILLATOR_Initialize();
    INTERRUPT_Initialize();
    INTERRUPT_GlobalEnable();
}
```



■ **SCREENSHOT 1.** The MPLAB Code Configurator is very easy to use. If you manipulate the Code Configurator correctly, it will properly write most of your PIC32MM0064GLP028 code.

The Code Configurator generated the files *pin_manager.c*, *pin_manager.h*, *interrupt_manager.c*, and *interrupt_manager.h*. In that we have not wandered into GPIO or interrupt territory, these files are mere skeletons at this point. All of the initialization function code alluded to in the *SYSTEM_Initialize* function is contained within the associated *.c* source file.

UART Activation

Setting up a PIC32MM0064GLP028 UART is as easy as clicking on check boxes. The MPLAB Code Configurator was used to generate the physical PIC32MM0064GLP028 results you see in **Screenshot 2**. On the outside, it looks like we used the Code Configurator to set up a simple 9600 bps UART. However, the *Enable UART Interrupts* check box contains a check and a monster was unleashed behind the scenes.

The MPLAB Code Configurator generated a *uart.c* and *uart.h* file. The Code Configurator also generated UART initialization code, multiple UART driver functions, and a UART interrupt handler. Here is a list of the UART functions generated by the MPLAB Code Configurator:

```
void UART1_Initialize(void);
uint8_t UART1_Read( void);
unsigned int UART1_ReadBuffer( uint8_t *buffer ,
    const unsigned int numbytes);
void UART1_Write( const uint8_t byte);
unsigned int UART1_WriteBuffer( const uint8_t
    *buffer , const unsigned int numbytes );
uint8_t UART1_Peek(uint16_t offset);
unsigned int UART1_ReceiveBufferSizeGet(void);
unsigned int UART1_TransmitBufferSizeGet(void);
bool UART1_ReceiveBufferIsEmpty (void);
bool UART1_TransmitBufferIsFull (void);
UART1_STATUS UART1_StatusGet (void );
```



```
UART1_TRANSFER_STATUS UART1_TransferStatusGet
(void );
```

At first glance, the function list is full of complicated code. If you study the *uart.h* file carefully, you will find that every aforementioned UART function includes an example of how to utilize the function. For instance, here is the code generated for the UART function *UART1_Write*:

```
char          myBuffer[MY_BUFFER_SIZE];
unsigned int   numBytes;
// Pre-initialize myBuffer with
// MY_BUFFER_SIZE bytes of valid data.
numBytes = 0;
while( numBytes < MY_BUFFER_SIZE);
{

    if( !(UART1_TRANSFER_STATUS_TX_FULL &
    UART1_TransferStatusGet()) )
    {
        UART1_Write(myBuffer[numBytes++]);
    }

    // Do something else...
}
```

The actual loading of the byte into the UART TX register occurs within the transmit interrupt. *UART1_TRANSFER_STATUS_TX_FULL* is defined within the *uart.h* file. If we were to not click the Code Configurator *Enable UART Interrupts* check box, things become much simpler. The function count is decreased dramatically:

```
uint8_t UART1_Read(void);
void UART1_Write(uint8_t txData);
UART1_STATUS UART1_StatusGet (void );
```

The non-interrupt UART read/write functions are so obvious that there are no associated examples. However,

the operation of the status function is made clear in the *uart.h* file example code:

```
while(!(UART1_StatusGet & UART1_TX_COMPLETE
))
{
    // Wait for the transmission to complete
}
```

The value of the status flag *UART1_TX_COMPLETE* — which is actually a bit value from the U1STA register — is defined within the *uart.h* file. The non-interrupt UART read and write functions are generated as follows:

```
uint8_t UART1_Read(void)
{
    while(!(U1STAbits.URXDA == 1))
    {
    }

    if ((U1STAbits.OERR == 1))
    {
        U1STAbits.OERR = 0;
    }
    return U1RXREG;
}

void UART1_Write(uint8_t txData)
{
    while(U1STAbits.UTXBF == 1)
    {
    }

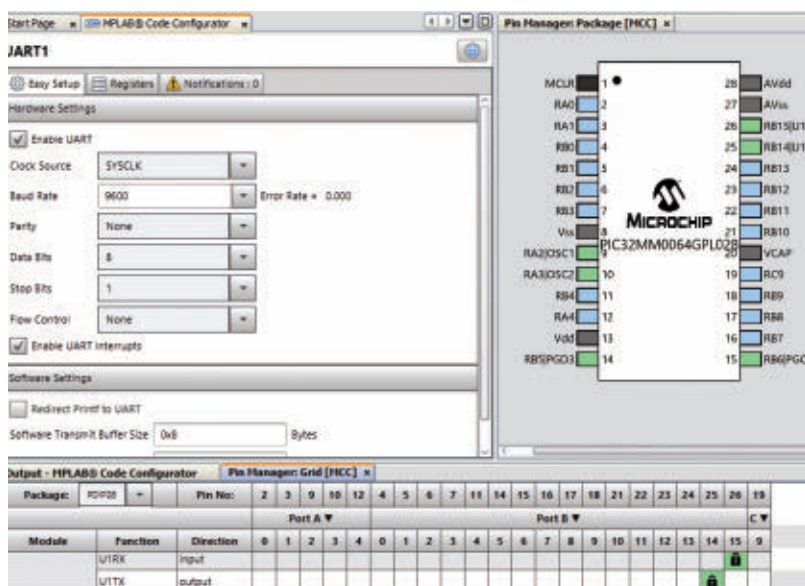
    U1TXREG = txData;
    // Write the data byte to the
    // USART.
}
```

Enough said.

Analog Processing with the MPLAB Code Configurator

All you need to know about how the analog-to-digital channel (ADC) is set up can be found in **Screenshot 3**. We have configured a 10-bit single-channel ADC that is referenced to the PIC32MM0064GLP028's Vdd and Vss power plane. The ADC clock is taken from the PIC32MM0064GLP028's internal free-running RC oscillator. Choosing the system peripheral clock (24 MHz) to drive the ADC does not allow a TAD time that falls within the minimum limits of the ADC's acquisition parameters. The MPLAB Code Configurator posts a Notification (Notifications tab) to warn against this TAD discrepancy. Once the FRC clock source is selected, the Notification is removed.

Once the ADC parameters are defined and loaded into the MPLAB Code Configurator, we can select an analog input pin. As you can see in **Screenshot 4**, we have brought the Code Configurator's pin module to the forefront. All of



SCREENSHOT 2. This capture seems to indicate that we used the MPLAB Code Configurator to set up a simple UART that runs at 9600 bps. Under the covers, things are a bit more complex.

the pin assignments for the PIC32MM0064GLP028 peripherals that we have previously activated are displayed. The most recent pin assignment is AN0, which happens to be the pin we selected for our ADC analog input. The name of pin RA0 (AN0) is user selectable and is named within the pin module window shown in **Screenshot 4**.

Our `SYSTEM_Initialize` function list is growing with the addition of the ADC peripheral to our project mix:

```
void SYSTEM_Initialize(void)
{
    PIN_MANAGER_Initialize();
    OSCILLATOR_Initialize();
    INTERRUPT_Initialize();
    ADC1_Initialize();
    UART1_Initialize();
    INTERRUPT_GlobalEnable();
}
```

In addition, two new files — `adc1.c` and `adc1.h` — have been automatically added to our MPLAB X PIC32MM0064GLP028 project. A complement of ADC driver functions was also generated:

```
void ADC1_Initialize (void);
void ADC1_Start(void);
void ADC1_Stop(void);
uint16_t ADC1_ConversionResultBufferGet(
    uint16_t *buffer);
uint16_t ADC1_ConversionResultGet(void);
bool ADC1_IsConversionComplete( void );
void ADC1_ChannelSelect( ADC1_CHANNEL
    channel );
void ADC1_Tasks(void);
```

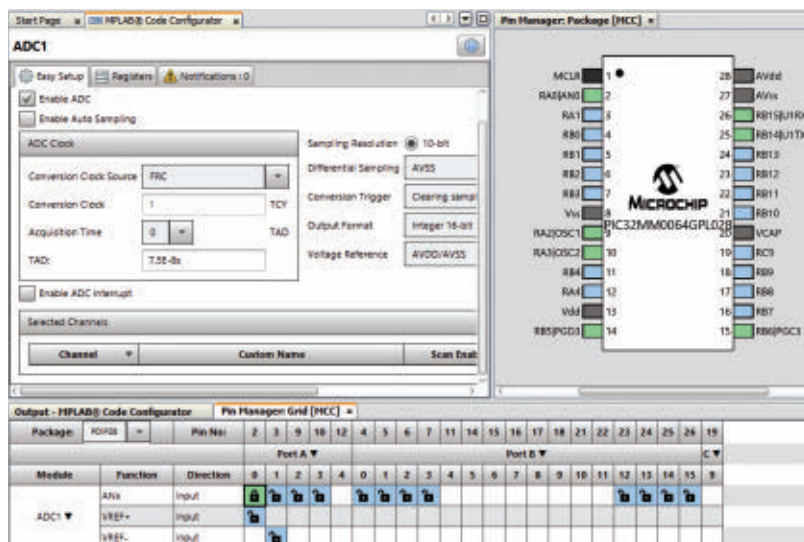
As you would expect, the MPLAB Code Configurator kindly provided some example code. Here's a generic ADC sample sequence that can be found within the text area of `adc1.h`:

```
int conversion;
ADC1_Initialize();
ADC1_ChannelSelect(AN1_Channel);
ADC1_Start();
//Provide Delay
for(int i=0;i <1000;i++)
{
}

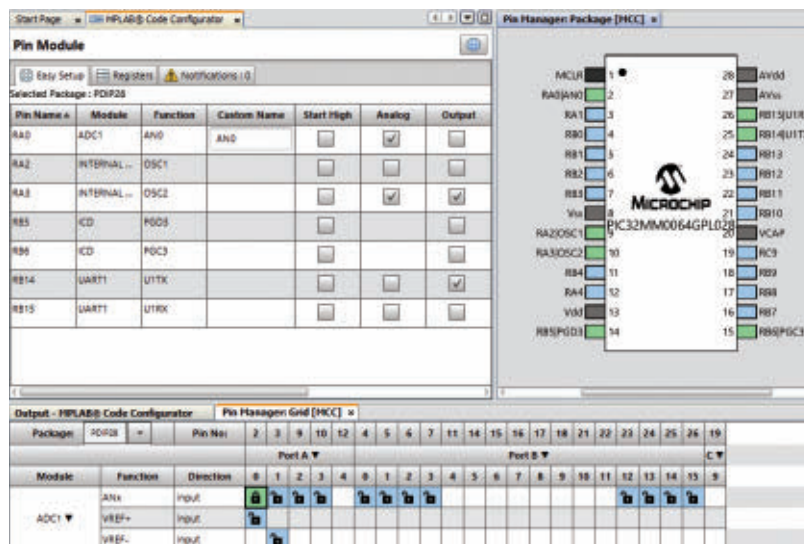
ADC1_Stop();
while(!ADC1_IsConversionComplete())
{
    ADC1_Tasks();
}
conversion = ADC1_ConversionResultGet();
```

Most of the ADC driver functions are short and sweet. Added sweetness is that you didn't have to write one single line of code to implement them:

```
void ADC1_Start(void)
```



■ **SCREENSHOT 3.** Setting up an ADC is just a couple of clicks of finger work. Once the ADC parameters are fed into the MPLAB Code Configurator GUI, an analog input pin can be selected from the Code Configurator pin module.



■ **SCREENSHOT 4.** The Pin Manager display is always accessible. Once you select a pin in the Pin Manager or via a peripheral assignment, you can transition to the pin module window to configure the selected pin's behavior and human-readable name.

```
{
    AD1CON1bits.SAMP = 1;
}
void ADC1_Stop(void)
{
    AD1CON1bits.SAMP = 0;
}
uint16_t ADC1_ConversionResultGet(void)
{
    return ADC1BUF0;
}

bool ADC1_IsConversionComplete( void )
```

```
{
    return AD1CON1bits.DONE;
    //Wait for conversion to complete
}
```

Up to this point, we have produced PIC32MM0064GLP028 start-up code and implemented a UART and ADC. Thus far, we have not written a single line of code.

GPIO Configuration Using the MPLAB Code Configurator

Our original schematic is rather bare. So far, we added a UART interface on pins RB14 and RB15. We have also parked an analog input on RA0. What if we wanted to hang an LED on RB13 and a pushbutton input on RB12? If we use the Code Configurator to perform the LED and button GPIO assignments, we won't even break a sweat.

Let's start with the LED. The story is told in **Screenshot 5**. We selected a free pin (in this case, RB13) in the pin manager window. Note that the pin can be selected as an input or output. Since we're driving an LED, the choice is output pin. Once the pin is selected, we can finish the pin's configuration in the pin module window. As you can see in **Screenshot 5**, we've named the pin LED, which is also reflected in the PIC32MM0064GLP028 graphic.

Configuring the pushbutton input pin follows a similar process. Instead of selecting RB12 as an output in the pin manager, we click on the input lock. You can see that I've done just that in **Screenshot 6**.

The MPLAB Code Configurator is an equal opportunity code generator. So, what it does for

peripherals, it also does for pins. Here is the driver code for pins AN0, LED, and BUTTON:

```
#define AN0_SetHigh()      LATABits.LATA0 = 1)
#define AN0_SetLow()      (LATABits.LATA0 = 0)
#define AN0_Toggle()      (LATABits.LATA0 ^= 1)
#define AN0_GetValue()    PORTABits.RA0
#define AN0_SetDigitalInput()
                        (TRISABits.TRISA0 = 1)
#define AN0_SetDigitalOutput()
                        (TRISABits.TRISA0 = 0)
#define BUTTON_SetHigh()  (LATBbits.LATB12 = 1)
#define BUTTON_SetLow()   (LATBbits.LATB12 = 0)
#define BUTTON_Toggle()   (LATBbits.LATB12 ^= 1)
#define BUTTON_GetValue() PORTBbits.RB12
#define BUTTON_SetDigitalInput()
                        (TRISBbits.TRISB12 = 1)
#define BUTTON_SetDigitalOutput()
                        (TRISBbits.TRISB12 = 0)
#define LED_SetHigh()     (LATBbits.LATB13 = 1)
#define LED_SetLow()      (LATBbits.LATB13 = 0)
#define LED_Toggle()      (LATBbits.LATB13 ^= 1)
#define LED_GetValue()    PORTBbits.RB13
#define LED_SetDigitalInput()
                        (TRISBbits.TRISB13 = 1)
#define LED_SetDigitalOutput()
                        (TRISBbits.TRISB13 = 0)
```

The pin driver definitions are obvious to the most casual observer. However, the Code Configurator still generates examples in the *pin_manager.h* file just as it did for the ADC and UART peripherals. Here's an example of how to obtain the BUTTON pin's logical status:

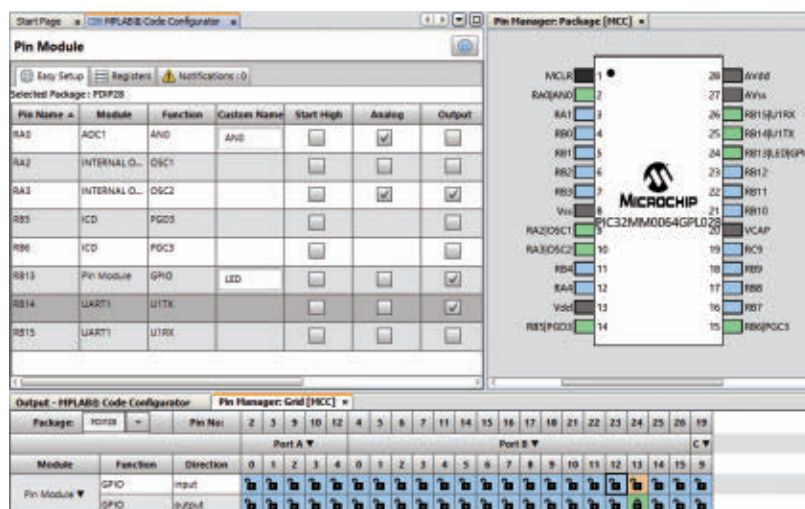
```
uint16_t portValue;

// Read RB12
portValue = BUTTON_GetValue();
```

Timing is Everything

No matter how many PIC32MM0064GLP028 peripherals or GPIO pins we configure, sooner or later our application will require the services of a timer. Normally, the timer is used as a periodic interrupt time base. Configuring a timer to overflow and trigger an interrupt on a periodic basis is a walk in the park when we include the MPLAB Code Configurator in the process. The interrupt period specified in **Screenshot 7** is 10 mS.

When using the Code Configurator, no manual computations are necessary to get the correct timer overflow value for a 10 mS periodic interrupt. Selecting a prescaler value that allows the timer period limits to fall into your interrupt period requirements is all that you need to do. As you can see in **Screenshot 7**, a prescaler value of 8 will allow a 10 mS timer period. The overflow value you see in the Period Count window is automatically calculated by the Code Configurator.



■ **SCREENSHOT 5.** This is how it works. Pick out a free PIC32MM0064GLP028 pin, declare it output or input in the Pin Manager window, set the pin properties — including the pin name — in the pin module window and click the Generate button. The Code Configurator generates all of the drivers for the selected pin automatically and places the code in your project.

As automatic as things have been up to now, we have come to a point where we will have to do a little work. Here's the timer interrupt handler that was generated by the Code Configurator:

```
void __attribute__((vector(_TIMER_1_VECTOR), interrupt(IPL1SOFT)))
TMR1_ISR()
{
    /* Check if the Timer Interrupt/ */
    /* Status is set */

    /***User Area Begin

    // ticker function call;
    // ticker is 1 -> Callback function
    // gets called everytime this ISR
    // executes
    TMR1_Callback();

    /***User Area End

    tmr1_obj.count++;
    tmr1_obj.timerElapsed = true;
    IFS0CLR= 1 << _IFS0_T1IF_POSITION;
}
```

The timer interrupt handler code is executed every 10 mS. Let's suppose we want to toggle the LED every 500 mS. Our LED toggle code would have to reside within the timer callback function (*TMR1_Callback*):

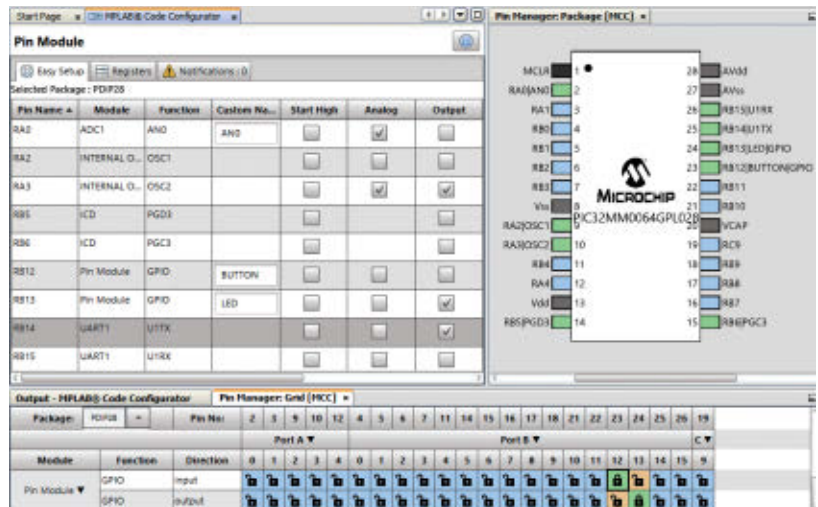
```
void __attribute__((weak)) TMR1_Callback(void)
{
    // Add your custom callback code here
}
```

We will need to declare a single-byte 10 mS counter variable. Let's call it *cntr10mS*. We'll use the LED driver and some standard C stuff to whip up a bit of code that


will blink the LED. When it's all said and done, our callback function looks like this:

```
uint8_t cntr10mS;
void __attribute__((weak)) TMR1_Callback(void)
{
    if(++cntr10mS == 50)
    {
        cntr10mS = 0;
        LED_Toggle();
    }
}
```

If that's too much work for you, we can skin this cat another (and even easier) way. I've specified a Callback




■ **SCREENSHOT 6.** In the pin module and Pin Manager universe, input pins get the same treatment as output pins. A free PIC32MM0064GLP028 pin is selected, declared as an input in Pin Manager, and configured in the pin module. Pin drivers are generated following a click on the MPLAB Code Configurator Generate button.

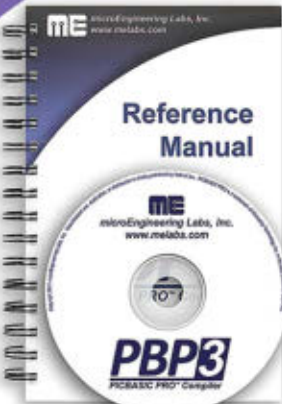


www.melabs.com


Development Tools and Learning Systems

PICmicro® microcontrollers



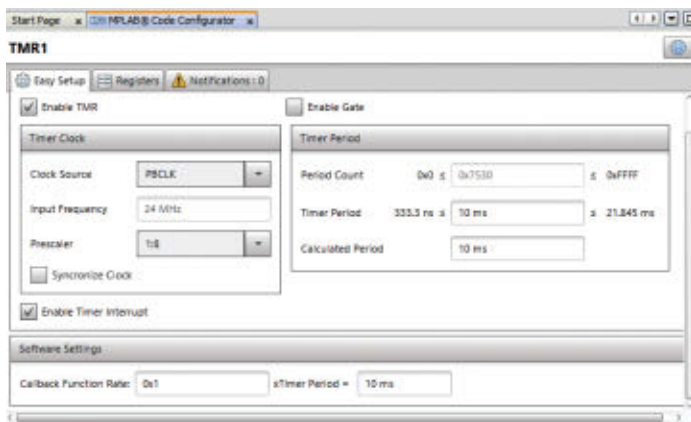


- PICBASIC PRO™ Compiler 3.0
- M.E. Labs Trainer
- U2 Programmer
- Prototyping Boards
- Experimenter Boards



Special Price of \$149.99
Use Coupon Code NV916 at checkout
 Valid until September 30, 2016
 (Retail price \$189.99)
 Academic and bulk pricing available

PICBASIC and PICBASIC PRO are trademarks of Microchip Technology Inc. in the USA and other countries. PIC is a registered trademark of Microchip Technology Inc. in the USA and other countries.



■ **SCREENSHOT 7.** We don't have to work hard here. However, we will need to craft a line or two of code to service the timer interrupt.

Function Rate value of 0x32 in **Screenshot 8.** Our timer interrupt handler now takes this form:

```
void __attribute__((vector(TIMER_1_VECTOR),
interrupt(IPL1SOFT))) TMR1_ISR()
{
    /* Check if the Timer Interrupt/Status is */
    /* set */

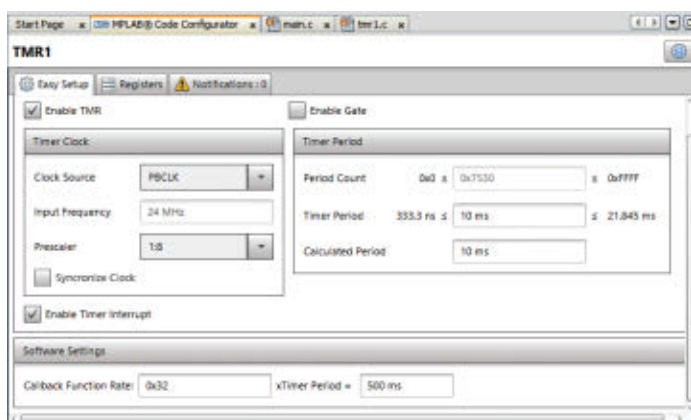
    /***User Area Begin
    static volatile unsigned int CountCallBack
    = 0;

    // callback function - called every 50th
    // pass
    if (++CountCallBack >= TMR1_INTERRUPT_
    TICKER_FACTOR)
    {
        // ticker function call
        TMR1_CallBack();

        // reset ticker counter
        CountCallBack = 0;
    }

    /***User Area End

    tmr1_obj.count++;
    tmr1_obj.timerElapsed = true;
    IFS0CLR= 1 << _IFS0_T1IF_POSITION;
}
```



■ **SCREENSHOT 8.** We can force the MPLAB Code Configurator to perform the 10 mS loop counting by entering a Callback Function Rate value.

Note the addition of a callback counter declared as *CountCallBack*. The Code Configurator has also generated a new definition:

```
#define TMR1_INTERRUPT_TICKER_FACTOR    50
```

Now our callback function looks like this:

```
void __attribute__((weak)) TMR1_CallBack(void)
{
```

The bottom line is that both timer interrupt schemes work. We just let the Code Configurator do a bit more math work in the latter example.

The Best of Both Worlds

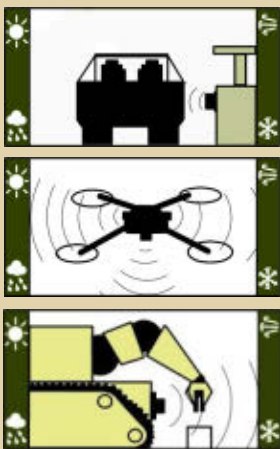
With the MPLAB Code Configurator treading in 32-bit territory, we indeed have the best of both worlds. As you have seen, the Code Configurator is an excellent code generator. You can use the Configurator-generated code in its original form or modify it to suit your needs.

We didn't write much code this month and we didn't spend any money. MPLAB X, the MPLAB Code Configurator, and the XC32 C compiler are free for the download. That makes them perfect additions to your Design Cycle. **NV**

CLASSIFIEDS

ROBOTICS

Need sensors?



www.maxbotix.com

LIGHTING



HARDWARE WANTED

DEC EQUIPMENT WANTED!!!

Digital Equipment Corp.
and compatibles.
Buy - Sell - Trade

CALL KEYWAYS 937-847-2300
or email buyer@keyways.com

Game Time

I'm writing from the road again. This trip has me in my old stomping grounds of the Dallas-Fort Worth Metroplex. After driving a colleague to the airport, I popped into one of my favorite places in the area: Tanner Electronics. I'll need parts while I'm here, and it's great to have a source run by such lovely people. After that stop, I continued up I35 to Lewisville where I visited my friends at Escape Rooms HQ. They've only been open a few months and are booked solid — so much so that they're opening a second location. ERHQ uses Propellers (coded by yours truly) in several props and puzzles, so I'm very happy that things are going well. Their timing for creating a well-executed escape room seems to have been spot on.

Escape rooms are all about timing: You're allotted a specific amount of time (typically, an hour) to solve the puzzles and escape. Most escape room businesses use a computer for overall game timing, but what if we want to create something small and stand-alone; something we could play at a party? Over the past few issues, we have explored all the elements to create such a prop. So, this time, we'll knit them together to create a timed game piece that can be reconfigured from its own interface.

The Display's the Thing

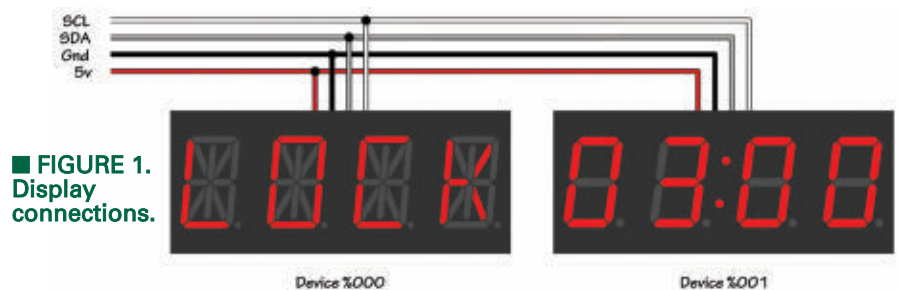
I'm using the same circuit from the July column, with the addition of another display module from Adafruit. The great thing about I²C is pin sharing; we can connect the second display right to the first (**Figure 1**). The second display has four seven-segment digits and includes a colon in the middle which makes it perfect for displaying time.

Yes, we can display time on the 14-segment display, but it's not ideal without the colon. Humans are very visual and prefer little clues like a colon which is used when formatting standard time.

Before we get to the displays, let's get to the heart of them. Both displays use the Holtek HT16K33. This chip will control up to eight digits with up to 16 segments each.

It also has the ability to scan keys, but that feature is not made available in the Adafruit displays, so we'll skip it.

The interface is through I²C and is darn easy. The `start()` method for the base object

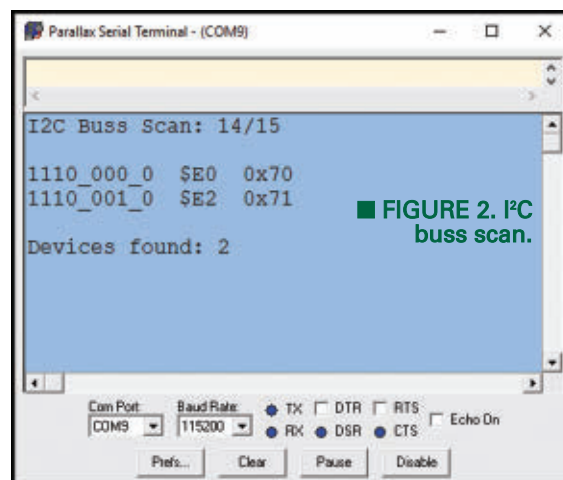


■ **FIGURE 1.**
Display
connections.

allows us to specify the I²C pins and the device address (%000...%111). The device address is combined with the HT16K33 slave ID (%1110_000_0) so that we target the correct display when more than one is on the same buss. If we run an I²C buss scanner (included in article downloads), we'll see the independent addresses of the displays (**Figure 2**).

There's only a few control registers that we have to be concerned with; most of our work will be writing segment data to the display. On power-up, the HT16K33 oscillator is stopped and the display is blank. The system setup register (\$2x) uses BIT0 to enable the oscillator. If we write \$21 to the display, the oscillator will start running. The display control register (\$8x) allows us to set the display

on or off, as well as control the blinking. Blinking uses two bits which gives us none, 2 Hz, 1 Hz, or 0.5 Hz blink rates. The object keeps a copy of this byte so that one element can be changed without losing the other. We can choose between the `set_display()`, `display_on()`, and `set_blink()` methods to affect the display control register:



■ **FIGURE 2.** I²C
buss scan.

Jon "JonnyMac" McPhalen
jon@jonmcpalen.com

Parallax, Inc.
www.parallax.com


```
pub set_display(state, rate)

  if (state)
    state := 1

  if ((rate < BLINK_OFF) or (rate >
    BLINK_0x5HZ))
    rate := BLINK_OFF

  dispctrl := $80 | (rate << 1) | state

  write_cmd(dispctrl)

pub display_on(state)

  if (state)
    dispctrl |= %0000_0001
  else
    dispctrl &= %1111_1110

  write_cmd(dispctrl)

pub set_blink(rate)

  if ((rate < BLINK_OFF) or (rate >
    BLINK_0x5HZ))
    rate := BLINK_OFF

  dispctrl &= %1111_0001
  dispctrl |= rate << 1

  write_cmd(dispctrl)
```

Finally, the display brightness can be set to one of 16 levels by writing \$E0 (1/16th brightness) to \$EF (full brightness). The **set_brightness()** method takes care of the details:

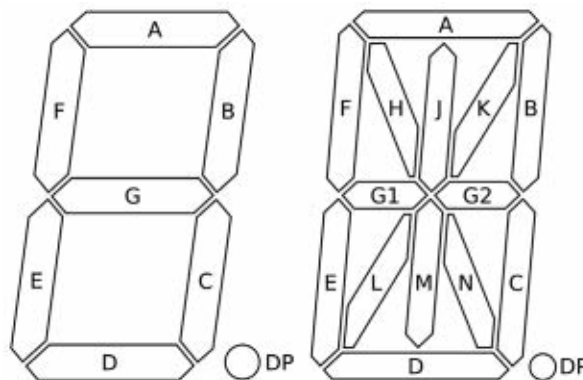
```
pub set_brightness(b)

  write_cmd($E0 | (0 #> b <# 15))
```

Segments for a column are updated by writing them to the address for that column. The HT16K33 controls 16 rows, so each column will get two bytes for a complete character. The base address for each column is the column number (0..7) multiplied by two.

I used the HT16K33 core object to create high-level objects for each of the displays. The primary difference between them is the number of segments (14 versus 7), which means that we have different character maps, and some features are better suited for one display than the other. Still, I tried to keep the objects as consistent with each other as I could. For those interested in creating custom character maps, **Figure 3** shows the segment assignments for both types of displays.

In truth, the 14-segment driver is the easier of the two to use. The Adafruit #1911 module uses four 14-segment displays with decimal points.



■ FIGURE 3. Display segments.

These are connected to columns 0..3 of the HT16K33. Writing any character is a breeze, and the object even includes standard string and numeric methods (liberated [with modifications] from FullDuplexSerial).

The display that looks simpler — the Adafruit #875 — is actually a tad trickier to use. The LED module is a four-digit seven-segment display with decimal points for each digit. This display was intended to be used in clocks as well, so it includes a colon between the second and third digits. In my idea of an easy world, the colon would have been mapped to column four (the fifth column), but it is, in fact, mapped to column two (third column, which matches its physical placement). This became a bit of a snag because my string and numeric methods expect the four characters to be contiguous — now we have a break. I don't tend to use **lookup** in Spin, but in this case it became very handy. What I needed was a quick way to convert the sequence 0 1 2 3 4 to 0 1 3 4 2. Here's the simple method that does that:

```
pri remap(c)

  if ((c => 0) and (c <= 4))
    return lookupz(c : 0, 1, 3, 4, 2)
  else
    return c
```

This lets us pass the virtual column for the seven-segment display and get the correct physical column back. We could have used a case structure, too, but I think using **lookup** is the most elegant solution. In any place where we need to specify a column, the **remap()** method sorts things out. Excellent. Now we have two fully featured LED displays and we can connect them to the same buss — but only after we change the address of one. For the project, I put a solder blob on the A0 pads of the seven-segment display; this gives it a device address of one for the HT16K33.

Author's Note: While nearly done with this column, I popped back into Tanner's and found another Adafruit display (#1270) which is also four by seven segments. This display uses large digits (1.2"), has two colons (the leftmost has individually-controlled dots), and a dot which I believe is used in temperature displays. I have updated the `jm_ht16k33_4x7segs.spin` so that features of the small and large displays are easily accessed. Either display will work with the program presented here.

Game On!

In the May issue, we created a background cog that includes a running milliseconds

Item	Source	BOM
Propeller Activity Board	Parallax #32910	
4x4 Matrix Keypad	Parallax #27899	
1/4W 1K resistor	Mouser #291-1K-RC	
1/4W 10K resistor	Mouser #291-10K-RC	
I ² C Alpha/Num Display	Adafruit #1911	
I ² C Seven-Segment Display	Adafruit #878	
M/M jumpers (w/o housing)	Polulu #1802	
M/F jumpers (w/o housing)	Polulu #1801	
4x1-pin housing	Polulu #1903	
1x1-pin housing	Polulu #1900	

register. In July, we created a simple electronic lock using a 4x4 matrix keypad. For an escape room type prop/game, we're going to combine these elements, and – as we should – make reconfiguration very easy.

Before we jump all the way in, let's talk about features; knowing what we want the program to do *before* we start coding will save a lot of time in the end:

- * Count down in one second intervals
- * Allow the user to enter a four-character code to unlock/disable
- * Time-out of current code entry if user takes too long
- * Prompted configuration

We already have a one millisecond background process, so adding a countdown timer to it is a breeze. We'll also add support for starting, stopping, and resuming the countdown. Here's the working part of the code that gets called from the background process loop:

```
var
    long    gmillis
    long    gsecs
    long    gmins
    long    grunning

pri game_timer
    if (grunning)
        if (++gmillis == 1000)
            gmillis := 0
            if (-gsecs < 0)
                if (gmins)
                    gsecs := 59
                    -gmins
                else
                    longfill(@gsecs, 0, 3)
```

Assuming there are values in the *gsecs* and *gmins* time registers, the method will operate when the *grunning* flag is true. At the top, we increment *gmillis* and compare it to 1000 to determine if a second has elapsed. When it has, *gmillis* is reset to 0, and the *gsecs* register is decremented. If *gsecs* rolls under (to -1), we check the *gmins* register for a value. If there are still some minutes left, *gsecs* is set to 59 and *gmins* is decremented. If there is a roll-under of *gsecs* when *gmins* is at zero, the timer has expired and we shut it down by writing 0 to the *gsecs*, *gmins*, and *grunning* registers. To simplify timer use, we have a few interface methods:

```
pub set_game_timer(gm, gs)

    longfill(@gmillis, 4, 0)
    gmins := gm // 100
```

```
gsecs := gs // 100

pub start_game_timer

    gmillis := 0
    grunning := true

pub stop_game_timer

    grunning := false

pub resume_game_timer

    if ((gsecs > 0) or (gmins > 0))
        grunning := true
```

You'll see that the **set_game_timer()** method takes two parameters, and allows them to be 0 to 99 (larger values are truncated with //). How can this be for a clock? Well, each half of the display can hold two digits, and we can write code to allow a funky time entry. While working on the program, I decided to enter a funky time into my microwave oven – it took it and dealt with it. We can, too. If the user enters 4:75 we will count down from there. The actual elapsed time will end up being 5:15, but we'll remain true to the user entry. The entry code and game time are store in a **DAT** table; this allows us to configure the default values in code, and then overwrite them in the EEPROM later. My habit is to keep most of the active code in methods so that the main loop is tidy and simply provides structure to the program:

```
dat
    GameCode      byte    "1234", 0
    GameTime      long    30_00

pub main

    setup

    check_update

    repeat
        reset_game
        update_display
        wait_key("#")
        start_game_timer
        repeat
            update_display
            if (grunning)
                check_user_key
            else
                if (gstate == G_OPEN)
                    time.pause(10_000)
                    wait_key("*")
                quit
```

At the top of the program, we must, of course, start our objects; this is done in the **setup()** method. The LED displays use different objects, but the core of those is the same (generic HT16K33 driver); hence, we must use separate addresses:

```
segs14.start(SCL2, SDA2, %000)
segs07.start(SCL2, SDA2, %001)
```

Here, you can see that the 14-segment display is using

HT16K33 addressed as %000, and the seven-segment display uses address %001. The prototype project is running on a Propeller Activity Board (PAB) which doesn't bring out the EEPROM I²C lines; hence, we have to define a secondary I²C buss. Let's assume the device is turned on and doesn't require reconfiguration. The master **repeat** loop runs and re-runs the game. The first step is to reset the game parameters:

```
pub reset_game

  set_game_timer(GameTime / 100, {
  } GameTime // 100)
  segs07.set_blink(segs07#BLINK_OFF)
  gstate := G_LOCK

  clear_code
  ktimer := millis - KEY_TIMEOUT
```

This bit of code explains why we store the game time as a decimal value in the format of HHMM; this facilitates entering a new run time using a standard decimal entry method. The **set_game_timer()** call takes care of extracting the minutes and seconds from the raw game time. Display blinking (from end of last play) is cleared and the game is set to lock. We then reset the user entry and start a key timer. Escape rooms are all about time and the psychological pressure created by time running out. The program has a feature that will clear an unfinished entry if the timeout expires — not unlike keyboard locks designed for doors, etc. This timer is using the background *millis* register as its reference:

```
pub update_display

  if ((millis - ktimer) < KEY_TIMEOUT)
    segs14.str(@keybuf)
  else
    if (gstate == G_LOCK)
      segs14.str(string("LOCK"))
      clear_code
    else
      segs14.str(string("OPEN"))

  segs07.clock(gmins, gsecs)

  if ((gmins == 0) and (gsecs == 0))
    segs07.set_blink(segs07#BLINK_2HZ)
```

The **update_display()** method does what is suggested based on conditions of the game. If a key has been pressed and the timeout has not expired, the key entry buffer is displayed. Otherwise, we see the game state (LOCK or OPEN) on the 14-segment module, with the current time on the seven-segment module. If time has run out, the clock module is set to blink. We are not using the *grunning* flag to determine the end-of-game; this allows us to add a pause/resume feature at some later point if we choose. With the initial display set up, the program waits for the [#] key to be pressed, starts the game timer running, and then drops into a secondary loop which is the active part of the game. At the heart of this loop is the **check_user_key()** method:

```
pub check_user_key | k

  ifnot (grunning)
    return

  k := key_to_ascii(read_keypad)

  case k
    "0".."9" :
      if (codelen < 4)
        keybuf[codelen++] := k
        ktimer := millis

    "*" :
      if (codelen > 0)
        keybuf[--codelen] := "*"
        ktimer := millis

    "#" :
      if (codelen == 4)
        if (strcmp(@GameCode, @keybuf))
          stop_game_timer
          gstate := G_OPEN
          ktimer := millis - KEY_TIMEOUT
```

This method checks the keyboard buffer for a key and deals with it. A number key will be added to the input buffer if there is space available. This also resets the key entry timer. We could expand this a bit to include the additional (alpha) keys on the 4x4 matrix, but not all alpha characters can be displayed on seven segments which we will use for setup; hence, we'll keep things clean by sticking to digits. The [*] key is used like the backspace to erase the last entry. Blank entries are marked with * in the display.

Finally, the [#] key tells the program where to compare our input with the unlock code in the program. If they match, the timer is stopped so you can boast about the amount of time we have left, and the game state is changed to OPEN. Regardless of the input, pressing [#] with four digits in the display will set the key entry timer to immediately timeout. If the entry is wrong, it gets erased, and the display goes right back to showing LOCK.

This is very simple yet effective. Plus, this code can be used in a wide variety of applications. I'm lucky in that I know a lot of people involved in the escape room business, so I get to write fun programs like this all the time. Not too long ago, I wrote a program that required two key switches to be turned simultaneously (nuclear bunker style) before the keyboard could be accessed. In that program, an incorrect entry locked the keyboard for several seconds (a time penalty). These are just layers on top of the simple timing code.

In the features list, there was the term "Prompted configuration" — that's handled in the **check_update()** method; this is an expansion of the same method in the electronic lock project from May:

```
pub check_update | idx, gtime

  time.pause(KEY_MS << 1)

  if (read_keypad <> STAR_POUND)
    flush_keypad
    return
```



```

segs14.scroll_str(@PromptCode, 250)
get_code(4)

byte move(@GameCode, @keybuf, 4)
repeat idx from 0 to 3
  ee.wr_byte(@GameCode[idx],
    GameCode[idx])

time.pause(500)
segs07.clear(true)

repeat
  segs14.scroll_str(@PromptTime,
    250)
  gtime := get_time
  if (gtime > 0)
    GameTime := gtime
    ee.wr_long(@GameTime, GameTime)
  quit

```

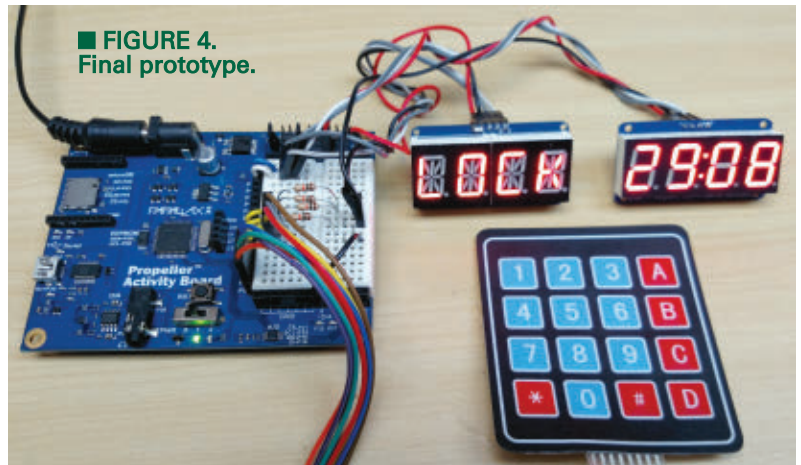
The **check_update()** method gets called at the very beginning of the program, so the first thing that happens is a delay to allow the keyboard to be debounced. If the keyboard does not return our special code for [*] + [#], we exit. The **scroll_str()** method is used to run a prompt message through the 14-segment module. As luck would have it, the last four letters of the message are “CODE,” and this remains in the display until a new four-digit code is entered. Likewise, for the game time, “TIME” will be displayed for that entry. The time value is accepted in a **repeat** loop so that we can put limits on the time entry; in this case, it must be greater than zero. We could set an upper limit if we like, as well. If a limit is violated, the loop runs again which re-displays the prompt and waits for input.

So, there you have it: a simple, configurable, countdown timer with keyboard entry. I use this as a starting point for many escape room props, but this code lends itself to a wide variety of timed control applications. **Figure 4** shows the updated project pieces with the code running. The final step is to wire up a PAB overlay board (#32999) so that the connections stay in place. I can swap out the Activity board if needed. Oh, and I guess I had better call Rick and ask him if he’ll help me machine an enclosure for everything.

New Life for PropBASIC

Like many, I taught myself to program in BASIC using the Timex-Sinclair 1000 computer. When the BASIC Stamp arrived in late 1993, I was thrilled because it meant I could apply my BASIC programming skills to the embedded projects I enjoy building. I finally bought my first BASIC Stamp 1 in early 1994, and I have been programming BS1 modules nearly every day since. Yes, I still use them.

It’s a long way from the BASIC Stamp 1 to the Propeller P1, but it doesn’t require a lot of scrutinizing to find the genetic similarities — they were born of the same creator (Chip Gracey). It makes sense, then, that a BASIC language in



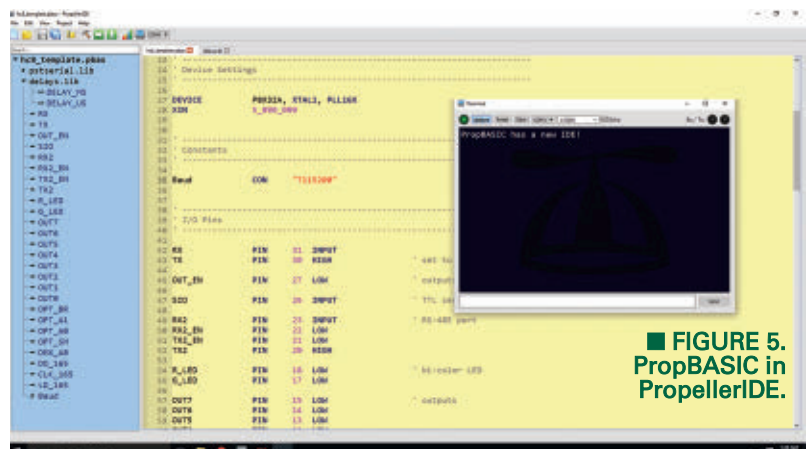
■ **FIGURE 4.**
Final prototype.

some form is available for the Propeller. PropBASIC is a derivative of SX/B, a compiler created by Terry “Bean” Hitt. I worked with him on SX/B while I was at Parallax, and that compiler (which was integrated into the SX-Key IDE [integrated development environment]) helped a lot of people migrate their BASIC Stamp projects to the SX. In the early days of EFX-TEK, all of our products were based on the SX and programmed in SX/B. It really was a wonderful tool for those of us who don’t enjoy programming in assembly language.

PropBASIC, though, struggled in the early days because it’s a compiler in need of an IDE, and there was never an ideal solution — until now. Brett Weir has been doing tremendous work on PropellerIDE, and his latest version enables seamless use of PropBASIC. This means that we can do BASIC and Spin programming side-by-side in the same editor! Nice! **Figure 5** is a screen capture of the latest PropellerIDE with a PropBASIC program loaded. Note how the project explorer panel details everything, with external files shown as collapsible lists. There are icons for subroutines, functions, variables, and even I/O pins.

If, like me, you have stayed away from PropBASIC because it lacked a proper IDE, those days are over.

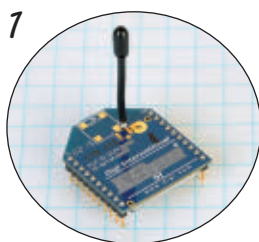
In BASIC or Spin (or even C or Forth), keep spinning and winning with the Propeller! **NV**



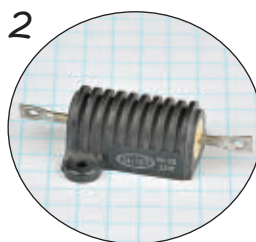
■ **FIGURE 5.**
PropBASIC in
PropellerIDE.

Name that Part!

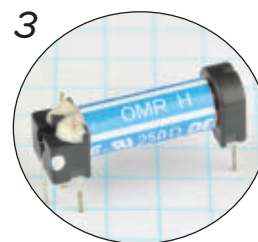
Try this photo quiz and see how many parts you can identify. Some parts date back to the 1950s and earlier, while others can be found at your local RadioShack. For scale, the blue background grid contains 1/4 inch squares. Keep in the mind the photos have been sized to fit the layout. The correct answers can be found on page [66](#). Good luck!



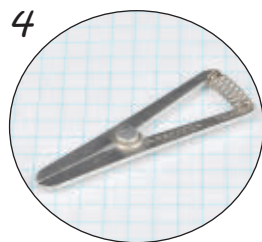
- a. GBe Transceiver
- b. XBee Transceiver
- c. Bumblebee Xmtr



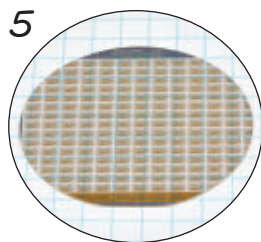
- a. CPU Heatsink
- b. Wire Guide
- c. Power Resistor



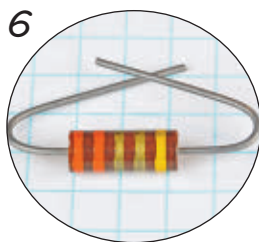
- a. 250 mH Inductor
- b. Reed Relay
- c. OMR Shunt



- a. Soldering Heatsink
- b. Alligator Clip
- c. Mini Gator Clip



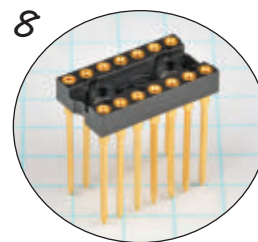
- a. Circular PCB
- b. Filter Grid
- c. Silicon Wafer



- a. 3.3 meg Resistor
- b. 3.3 ohm Resistor
- c. 33 ohm Resistor



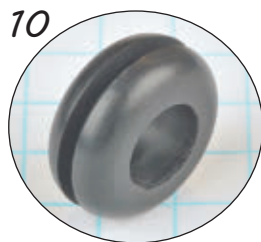
- a. 1/8" Stereo Plug
- b. 1/8" Mono Plug
- c. 1/4" Stereo Plug



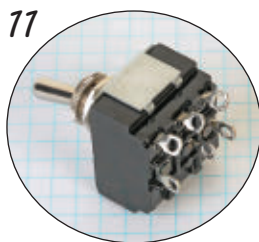
- a. Wirewrap IC Socket
- b. 16-pin IC Socket
- c. 14-pin DIP Shield



- a. RCA Phono Jack
- b. RG-59 Jack
- c. F-Type Jack



- a. Plastic Cable Guide
- b. Rubber Grommet
- c. Rubber Pulley



- a. DPDT Switch
- b. 3PDT Switch
- c. 3PST Switch



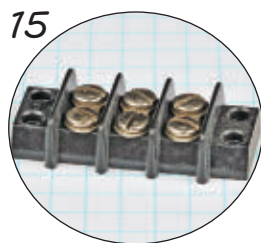
- a. HDMI Plug
- b. S-Video Plug
- c. Micro USB Plug



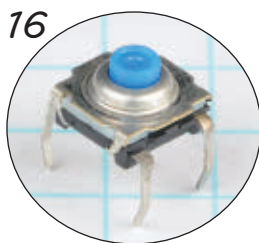
- a. 10-Turn Pot
- b. Combo Switch
- c. FM Dial Tuner



- a. Radio Tube
- b. TV Tube
- c. Audio Tube



- a. Barrier Strip
- b. Fanning Strip
- c. Six Screw Trimmer



- a. Blue LED
- b. Tactile Switch
- c. Potentiometer



- a. Type N Socket
- b. BNC Socket
- c. UHF Socket



- a. IC Puller
- b. Desoldering Tool
- c. Wire Stripper

Presented by David Goodsell

Scoring: 0-7 = Novice 8-15 = Good 16-18 = Expert

PRESENTED BY BARNES & NOBLE

WORLD



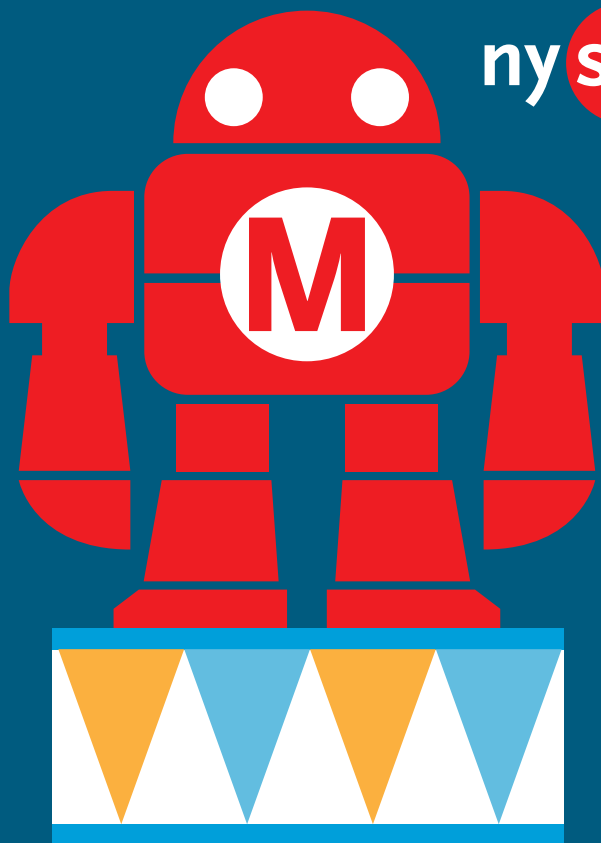
NEW YORK
HALL OF SCIENCE

7TH ANNUAL

Maker Faire®

ny sci

GREATEST
Show
& TELL
ON
Earth



OCT 1+2 NEW YORK HALL OF SCIENCE
FLUSHING MEADOWS, QUEENS, NY



PRESENTING SPONSOR

BARNES & NOBLE

GOLDSMITH SPONSORS

Google



SILVERSMITH SPONSORS



Makeblock

MICROCHIP
Atmel

GET YOUR
TICKETS
TODAY!



makerfaire.com

BROUGHT
TO YOU BY

Make:

>>> QUESTIONS

I Got The X10 Blues

My X-10 home automation gear has been reliable for years but just recently has become intermittent. Is there a way to test for what might be interfering with the operation?

#9161 Malcom Williams
Camden, NJ

What's The Deal With Ultra Caps?

I am looking to experiment with "ultra capacitors" as a replacement for AA batteries. Is this possible to do and, if so, what kind of capacitors would be a good place to start?

#9162 Glen Ross
Marietta, GA

Cassette to MP3

I have a box of cassette audio tapes that I want to convert to MP3 format. What's the simplest way to do it? I have a Windows 10 PC and a Nakamichi CR-2A cassette deck.

#9163 Samantha Costa
Boise, ID

Transistor Training

I'm retired and re-learning electronics. I am confused about the differences between MOSFET and "regular" transistors. Is there a rule of thumb as to when/where/why you would use one over the other?

#9164 Arthur Bergerson
Latham, NY

>>> ANSWERS

[#6162 - June 2016]

Wi-Fi For The Pi

Could someone please tell me if the ESP8266.01 module is compatible with the Raspberry Pi? And/or where can I find a Wi-Fi and GPS module that are?

There is an instructable on how to use the ESP8266 with the Raspberry

Pi at www.instructables.com/id/Connect-an-ESP8266. However, I don't recommend that you use it because it communicates over the hardware UART. This will result in an extremely slow 115 kbps Internet connection, about 1/100th of the speed of a typical 10 mbps cable connection. It will also tie up the UART which is needed for the GPS.

If you have a Raspberry Pi model 3, you do not need to add a Wi-Fi adapter at all since one is already built into the board. Use a version of Raspbian dated March 2016 or later to avoid driver issues. For older boards, the simplest solution for Wi-Fi is to use a small USB adapter. I have had success with the Edimax EW-7811Un, available from Amazon and others for \$9. It supports 802.11b/g/n up to 150 mbps and it's supported by any recent version of Raspbian.

GPS is also straightforward. Most receivers use a simple async serial port and are pre-configured to transfer data at 4800 bps, no parity, eight data bits, one stop bit, though this can be overridden in software. Data is transferred using the NMEA (National Marine Electronics Association) protocol. Google "GPS protocol specification" for information about this text-only protocol made up of 'sentences' that are easily parsed to get time and location information. If you are technically adept, you can get a Ublox NEO-6M or similar 3.3 volt serial GPS module for around \$15 from eBay or Amazon, and wire it to the serial port. Using the 26/40 pin I/O connector port, 3.3V power is on pin 1, ground on pin 6, Tx on pin 8, and Rx on pin 10.

A more expensive but beginner-friendly solution is to use an extender board known as a HAT that requires no hardware interfacing. One such HAT is the "Adafruit Ultimate GPS HAT for the Raspberry Pi," adafruit.com product #2324. This board

includes a GPS receiver, internal antenna, Real Time Clock (RTC), and a holder for a CR1220 RTC backup battery. It also has a U.FL connector for an optional external antenna. This will only work with an original or model 2 Pi as changes to the hardware in the model 3 has made it incompatible with this HAT, at least for now. The original Pi and model 2 Pi work fine. This limitation likely exists with other GPS HATs. If you have a model 3, make sure the GPS HAT mfg. specifically claims support for it before you buy.

Mark Lewus
Denville, NJ

[#6164 - June 2016]

PD Race Timer

Does anyone have a schematic for a pinewood derby finish line race timer? I would prefer to build something that doesn't use a microcontroller, if at all possible.

#1 When my grandchildren were into Pinewood Derby racing, I built a "which came first" circuit (**Figure1**). The system has two 12 volt lamps mounted above the track at the end. The car that breaks the beam causes the lamp to light and locks out the other track. A tie is not possible.

A 12 volt center-tapped transformer supplies 5 VDC to the logic and 12 VAC to the lamps. The logic consists of NAND gates and set-reset flip-flops. When one FF is set, it energizes a relay that lights the lamp and simultaneously locks out the other channel.

A 38 or 40 kHz IR transmitter is mounted above the track at the end and the receiver is mounted under the track (or vice versa). You may be able to find a receiver at RadioShack, eBay, or Amazon. The receiver is 38 kHz but 40 kHz will work fine. I used reed relays that will work on five volts but any five volt relay will work. My

Send all questions and answers by email to forum@nutsvolts.com
or via the online form at www.nutsvolts.com/tech-forum

original system included a three digit timer, but 8 to 10 year olds are not interested in the time.

Russell Kincaid
Milford, NH

#2 Try this link: www.techlib.com/electronics/games.html. It might be useful with a little change in switches.

Brian Angel
Baton Rouge, LA

[#6165 - June 2016]

Dog Bark Detector

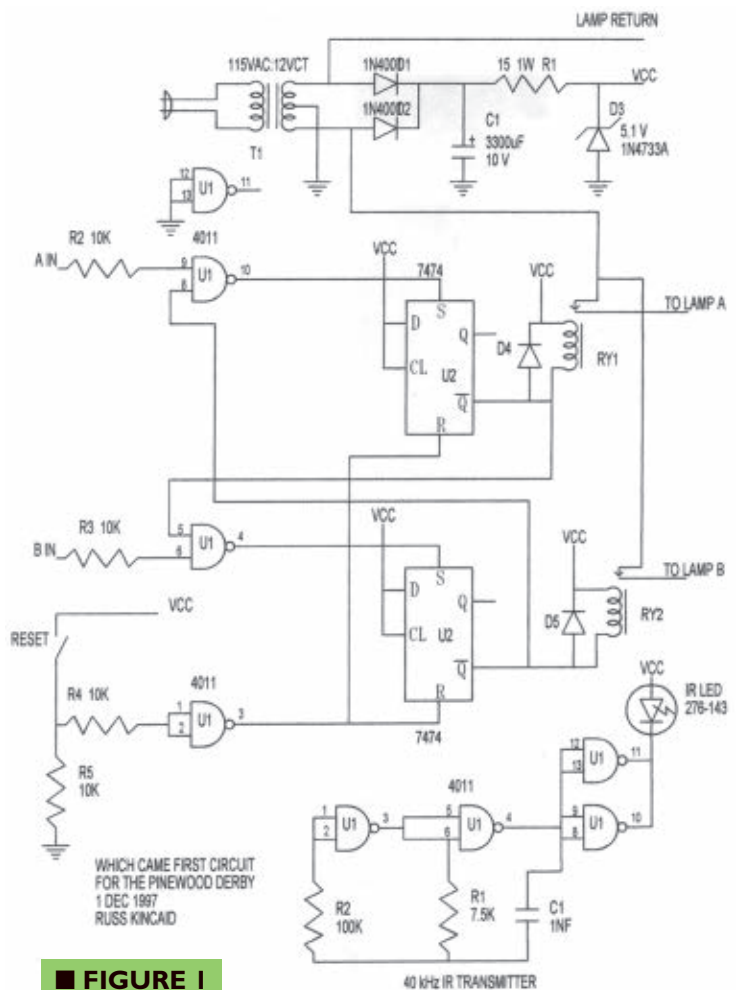
I have a very attentive dog who barks immediately when someone is in the backyard or at the front door. This is great for alerting me to visitors or possible intruders, but I really can't hear as well as I used to and my dog has a rather quiet and high-pitched bark that I mostly can't hear – especially if the TV is loud. I would like plans or a design for a “dog bark detector” to blink a light by my chair when the dog is barking so I don't miss these “alerts.” Does anyone have a circuit or schematic for such a thing?

You don't need to build hardware. You already have all the "hardware" you need. You need to do some programming; namely, programming (teaching) your dog to push on your hand when there is an intruder. Such a trained dog is known as a service hearing dog or signal dog.

Find a skilled trainer to teach your dog to alert you by physically pushing on your hand or if your dog is tiny to whack your ankle. A trained dog is infinitely more useful than any circuit you can build, e.g., when trained to do so, the dog will wake you up when the alarm clock goes off, keep you from being run over by cars and bicycles, tell you when people are sneaking up behind you, or are at the front door. A dog will alert you to anything you teach him/her.

The idea here — as in most engineering problems — is to solve the problem at hand in the simplest way, rather than forcing a preconceived and quite possibly inferior notion to do what you want.

Bruce Hartenbaum
via email



■ **FIGURE 1**

**Three Days.
Three Domains.
Countless Opportunities.**

UNMANNED SYSTEMS DEFENSE is an inclusive forum, bringing together industry, defense and government program managers, decision makers and technology experts for three intensive days of information sharing and networking. Each day is dedicated to a specific domain – maritime, air and ground – and includes panel discussions, in-depth presentations and networking opportunities.

REGISTER TODAY at thedefenseshow.org

**UNMANNED SYSTEMS
DEFENSE**
Powered by AUVSI

OCTOBER 25-27, 2016

The Ritz-Carlton, Pentagon City
Arlington, Virginia, USA

ELECTRONET

USB Add USB to your next project--
It's easier than you might think!

USB-FIFO • USB-UART • USB/Microcontroller Boards
RFID Readers • Design/Manufacturing Services Available
Absolutely NO driver software development required!
www.dlpdesign.com

Ironwood ELECTRONICS Innovative RF
Socket for QFN

No Mounting Hole, Tiny Footprint Socket

1-800-404-0204
www.ironwoodelectronics.com

PCB, PCBA and More! **Myro** Low cost
High Quality

www.myropcb.com
1-888-PCB-MYRO

ALL ELECTRONICS CORPORATION
Electronic Parts and Supplies.
www.allelectronics.com
Free 96 page catalog 1-800-826-5432

For the ElectroNet
online, go to
www.nutsvolts.com
click **Electro-Net**

superbrightleds.com
COMPONENT LEDS • LED BULBS • LED ACCENT LIGHTS

ANSWERS TO NAME THAT PART on page 62:

1b, 2c, 3b, 4a, 5c, 6b, 7a, 8a, 9c, 10b, 11c, 12a, 13a, 14b, 15a,
16b, 17c, 18c

ADvertiser INDEX

■ LOOK FOR ■ SEARCH FOR ■ FIND

Find your favorite advertisers here!

AUDIO/VIDEO

Team Kingsley23

BUYING ELECTRONIC SURPLUS

All Electronics Corp.40

CIRCUIT BOARDS

Acutrace67
AP Circuits6
ExpressPCB47
Front Panel Express LLC6
PCFabExpress22
Saelig Co., Inc.3

COMPONENTS

All Electronics Corp.40
Electronix Express21
Saelig Co., Inc.3
SDP/SI25

DESIGN/ENGINEERING/REPAIR SERVICES

Cleveland Institute of Electronics..7
ExpressPCB47
Front Panel Express LLC6

DEVELOPMENT PLATFORMS/TOOLS

AnarenBack Cover
M.E. Labs55
Technologic Systems2

EDUCATION

Cleveland Institute of Electronics..7
Command Productions23
M.E. Labs55
Poscope23

EMBEDDED SYSTEMS

Saelig Co. Inc.3
Technologic Systems2

EVENTS

AUVSI65
Maker Faire63

ENCLOSURES

Front Panel Express LLC6

LCDs/DISPLAYS

EarthLCD25
Saelig Co., Inc.3

MICROCONTROLLERS / I/O BOARDS

M.E. Labs55
Technologic Systems2

MISC./SURPLUS

All Electronics Corp.40

MOTORS / MOTOR CONTROL

Robot Power25
SDP/SI25

ROBOTICS

Cleveland Institute of Electronics..7
Electronix Express21
Robot Power25
SDP/SI25
Team Kingsley23

TEST EQUIPMENT

Electronix Express21
Poscope23
Saelig Co., Inc.3

TOOLS

PanaVise7
Poscope23

WIRE, CABLE AND CONNECTORS

All Electronics Corp.40

WIRELESS PRODUCTS

AnarenBack Cover
Technologic Systems2

Acutrace	67
All Electronics Corp.	40
Anaren	Back Cover
AP Circuits	6
AUVSI	65
CIE	7
Command Productions	23
EarthLCD	25
Electronix Express	21
ExpressPCB	47
Front Panel Express LLC	6
Maker Faire	63
M.E. Labs	55
PanaVise	7
PCFabExpress	22
Poscope	23
Robot Power	25
Saelig Co., Inc.	3
SDP/SI	25
Team Kingsley	23
Technologic Systems	2

PRINTED CIRCUIT BOARDS

THINK YOU CAN FIND PCB PRICES THAT BEAT OURS?

WE DARE YOU.

If you do, than we
will match the price
AND give you \$100
towards your
next order!

THERE ARE NO GAMES INVOLVED IN OUR PRICING



Our Capabilities:

- From same day quick turn prototype to production in under 10 days
- Full CAD and CAM review plus design rule check on ALL Gerber files
- Materials: Fr4, Rigid, Flex, Metal Core (Aluminum), Polyimide, Rogers, Isola, etc.
- HDI Capabilities: Blind/Buried Microvias, 10+N+10, Via-in-Pad Technology, Sequential Lamination, Any Layer, etc.
- Our HDI Advantage: Direct Laser Drilling, Plasma De-Smear Technology, Laser Microvia, Conductive Plate Shut.

Take the Accutrace Challenge and see WHY OUR PRICING CANNOT BE BEATEN

Accutrace inc.

www.PCB4u.com sales@PCB4u.com

● ITAR, ● I SO 9001:2008, ● UL Approved

"Fetch newspaper, Rex!"



A giant, coffee-carrying IoT dinobot at the command of your index finger? Why not. In fact, wherever your imagination leads you, Anaren empowers you with the industry's easiest-to-use wireless connectivity platform. Scan the code to learn more and register for a chance to win a free Atmosphere MSDK.* Think. Build. Connect.



Anaren®
What'll we think of next?®

* Must be 18 to participate. No purchase necessary. North America only. Void where prohibited. Other restrictions apply.

www.anaren.com/airforwiced/register-win